Modelling default risk including idiosyncratic default risk in returns on corporate bond portfolios

Liu, Hong

Awarding institution:
King's College London

Modelling default risk including idiosyncratic default risk in returns on corporate bond portfolios

Hong Liu

Department of Mathematics

King's College London

February 24, 2020

Abstract

Returns on corporate bond portfolios are subject to both systematic and idiosyncratic default risks. Koivu and Pennanen (2010) derived two-factor return models for corporate bonds, using one of the risk factors to approximate effects of systematic default risk on returns. Koivu and Pennanen (2014) similarly showed that modelling returns of index-linked bonds can be reduced to statistically modelling of two risk factors, one of which is an underlying index and could be used to model default losses. However it wasn't discussed further how this can be done. Our goal is to model stochastically default losses on returns of corporate bonds using the underlying index in Koivu and Pennanen (2014), whilst considering effects of both systematic and idiosyncratic default risks. We first give a more precise economic meaning to the underlying index for corporate bonds. We then express returns of corporate bonds as a function of default losses, which in turn are a function of the underlying index. Instead of approximating default losses using historical yield spreads like Koivu and Pennanen (2010), we model them over time as a compound (inhomogeneous) Poisson gamma process. Parameters of our default losses model are easily estimated using the Maximum likelihood method. Default losses simulated using our model are reasonably close to historical default losses of the Bank of America Merrill Lynch US High Yield index. Most importantly, using our proposed default losses model, our two-factor return model for corporate bonds is suitable for both well-diversified and non well-diversified bond portfolios.

Table Of Contents

1 Introduction

Returns on corporate bonds are subject to both systematic and idiosyncratic default risks. While systematic default risk is the uncertainty inherent in the entire market or market segment, idiosyncratic default risk is the uncertainty specific to a company or an industry. Koivu and Pennanen (2010) derived two-factor return models for several classes of bonds, including fixed rate government bonds, inflation linked bonds and corporate bonds. While the first risk factor is yield-to-maturity for all bond classes, the second risk factor was used to capture effects of unique features of each bond class on returns. For corporate bonds, the second risk factor was used to approximate effects of defaults on returns due to systematic default risk, whilst the idiosyncratic default risk was ignored. This approximation was based on the assumption that when portfolio becomes infinitely large, the portfolio becomes well-diversified. Koivu and Pennanen (2010) further approximated this risk factor with yield spreads between government bonds and corporate bonds assuming the existence of a risk neutral measure.

Koivu and Pennanen (2014) similarly showed that modelling returns of index-linked bonds can be reduced to statistically modelling of the portfolio's yield-to-maturity and the underlying index, for which many statistically validated models are available. The economic meaning of the underlying index needs to be given per bond class. For fixed rate government bonds, the underlying index was simply zero; for inflation-linked bonds, the underlying index was used to represent the consumer price index. Though Koivu and Pennanen (2014) mentioned that the underlying index could be used to model default losses for corporate bonds, they didn't discuss further how to model default losses with this underlying index such that effects of defaults are captured.

We aim to model stochastically default losses on returns of corporate bonds using this underlying index in Koivu and Pennanen (2014), considering effects of both systematic and idiosyncratic default risks. To achieve this goal, we first use the underlying index in Koivu and Pennanen (2014) for corporate bonds to represent remaining fraction of all outstanding payments due to both systematic and idiosyncratic default risks. Then returns of corporate bonds are expressed as a function of time, yield-to-maturity and default losses, which is expressed as a function of the underlying index. Instead of applying approximation to the default losses like Koivu and Pennanen (2010), we model them over time as a compound (inhomogeneous) Poisson gamma process. Parameters of this default losses model are easily estimated using the Maximum likelihood estimation; default losses simulated using the proposed model replicates reasonably well the historical default losses of the Bank of America Merrill Lynch US High Yield Index. Most importantly, using our default losses model, our two-factor return model is suitable for both well-diversified and non well-diversified corporate bond portfolios.

This thesis is organised as follows. Section 2 presents a literature review; section 3 details how we express returns of corporate bonds as a function of yield-to-maturity and default losses; section 4 describes the stochastic model we propose for losses in returns on corporate bond portfolios due to both systematic and idiosyncratic default risk; section 5 illustrates how to estimate parameters of the proposed model using the Maximum likelihood estimation; section 6 presents estimated model parameters and simulation results; section 7 draws conclusions.

2 Literature review

There has been a vast amount of literature in default risk modelling. However literature on modelling default risk in returns on corporate bond portfolios has been limited.

Default risk has two components, one is the arrival risk concerning time of default, the other one is the magnitude risk concerning loss given default (LGD). There are two approaches in modelling the default time in the literature. One is the *structural approach*, beginning with Merton (1974) in which the default event occurs when the firm value falls short of debt value at debt maturity. Hull and White (1995) and Longstaff and Schwartz (1995) allowed default to occur before debt maturity should the firm value reaches a threshold. In this approach the evolution of the firm value is explicitly modelled. The structural approach is not easy to use due to the firm value being in general unobservable or difficult to obtain. The alternative approach, which is more widely used given its tractability, is the *reduced-form approach* adopted by Duffie and Singleton (1999), Jarrow, Lando, and Turnbull (1997) and Madan and Unal (1998) etc. With this approach default is an unpredictable event that is governed by a hazard-rate process. Lando (1998) extended this approach by introducing the Cox processes in modelling the occurrence of default events. Jarrow, Lando, and Yu (2005) introduced the concept of conditionally diversifiable default risk using the framework of Lando (1998) and discussed empirical implications of diversifiable default risk. Those reduced-form models mentioned earlier only allowed a single default per credit name. In Schönbucher (1998), the reduced-form intensity approach was used whilst multiple defaults per credit name were allowed.

On the magnitude risk of default, the default magnitude is random and has a pre-determined distribution in the Merton type models. Hull and White (1995) and Longstaff and Schwartz (1995) have a constant default magnitude. Madan and Unal (1998) models the conditional risk neutral density for the default magnitude. Jarrow, Lando, and Turnbull (1997) used a recovery rate that is exogenously given and depends on the seniority of the risky zero-coupon debt. Duffie and Singleton (1999) introduced the concept of constant fractional recovery of market value (RMV) at defaults[1] and compared the implication of this to alternative recovery-of-face value (RFV)[2] and recovery-of-treasury (RT)[3] in corporate bond valuation. This concept of fractional recovery of market value is subsequently used in many papers, such as Lando (1998) and Schönbucher (1998). In Schönbucher (1998) the default losses can be predictable and a marked point process was used to describe the double sequence comprised of both default time and loss.

Reduced-form models have been applied to model different types of spreads: corporate-Treasury spreads in Duffee (2005), LIBOR-swap spreads in Collin-Dufresne and Solnik (2001), swap-Treasury spreads in Duffie and Duffie and Singleton (1997), Liu et al. (2000) and sovereign yield spreads in Duffie, Pedersen and Singleton (2001). They have also been applied to analyse the default risk premium, or the expected return on defaultable bonds. Jarrow, Lando, and Yu (2005) show that the default risk premia is composed of two parts, one due to systematic default risk and the other idiosyncratic risk. Systematic default risk is the uncertainty inherent in the entire market or entire market segment and therefore cannot be reduced through portfolio diversification. Idiosyncratic risk is the uncertainty specific to a company or an industry and can be reduced through portfolio diversification. Jarrow, Lando, and Yu (2005) use the notion of "conditionally diversifiable" to capture the idea that once the systematic parts of default risk have been isolated, the residual parts represent idiosyncratic or firm-specific shocks that are uncorrelated across firms. Conditional diversification also implies that idiosyncratic risk is diversifiable in large loan portfolio. Consequently there would be no risk premium for idiosyncratic default risk as shown in Jarrow, Lando, and Yu (2005).

There has been some literature in studying expected returns of corporate bond portfolios. Jarrow (1978) examined the relationship between the bond's yield to maturity and the expected return; an alternative formula for the systematic risk of a bond was also discovered. Schönbucher (2002) showed that the expected return on defaultable bonds can be decomposed into three components, one component is the expected return on an otherwise identical default-free bond, the second component is the difference between the risk-neutral and the physical mean-loss rate, the third component is the systematic default risk due to variations in the default intensity. Driessen (2005) empirically decomposed the expected corporate bond return into interest rate, default, liquidity and tax factors.

However literature on modelling default risk in returns on corporate bonds has been limited. Ilmanen et al. (1994) showed that duration's ability in measuring risk in bonds decreases when it comes to corporate bonds. For the reason that there are both interest rate and default risk present in returns of corporate bond portfolios. Ilmanen et al. (1994) used default spreads to measure default risk in their two-factor model to explain excess returns of corporate bonds. His analysis showed that default spreads is statistically significant in explaining excess returns of corporate bonds and duration can accurately capture interest rate in returns. However convexity is insignificant in explaining excess returns. Koivu and Pennanen (2010) developed a two-factor return model for fixed-rate government bonds, inflation-linked bonds and corporate bonds. This return model was based on Taylor approximation of the logarithmic bond price with respect to time, yield to maturity and outstanding coupon/principle payments. For corporate bonds, the second risk factor was used to approximate effects of systematic default risk in returns only; this second risk factor was subsequently approximated using yield spreads between short-maturity corporate bond government bonds. Thus their two-factor return model is only suitable for well-diversified bond portfolios. Koivu and Pennanen (2014) developed two-factor return models for index-linked bonds. Unlike Koivu and Pennanen (2010) , their models are based on Taylor approximation of the logarithmic bond price with respect to time, yield to maturity and an underlying index, the economic meaning of which needs to be defined per bond class. Though it was mentioned the underlying index could be used to model default losses for corporate bonds, it wasn't discussed further how to model default losses using the underlying index such that effects of defaults on returns are captured.

[1] "Recovery of market value (RMV)" means that the recovery of a defaulted asset at default is a fraction of its pre-default market value.

[2] "Recovery-of-face value" means that the recovery of a defaulted asset at default is a fraction of its notional value.

[3] "Recovery-of-treasury" means that the recovery of defaulted claims is espressed in terms of the market value of equivalent default-free assets.

3 Returns on corporate bond portfolios

Based on Koivu and Pennanen (2014), we use the underlying index to represent a remaining fraction of outstanding payments of corporate bonds in the presence of counterparty risk. Then returns of corporate bonds are expressed as a function of time, yield to maturity and default losses due to both systematic and idiosyncratic default risks.

Consider a corporate bond portfolio with finite N outstanding payment dates at times $t_1 < t_2 < \cdots < t_N$. The contractual amount of its nth outstanding payment payable at time t_n for $n \in \{1, 2, \ldots, N\}$ is denoted by c_n. In events of defaults, the realised payments are less than or equal to contractual payments. Assuming that all future payments will be reduced by the same fraction due to defaults, the realised amount of the nth outstanding payment at time $t < t_1$ will be

$$c_{t,n} = I_t c_n, \; n \in \{1, 2, \ldots, N\} \tag{1}$$

where recovery index $I_t \in [0, 1]$ represents remaining fraction of portfolio's outstanding payments due to defaults up to time t. As there are likely more defaults at time $s > t > 0$, realised outstanding payments at time s should be equal or smaller than those at time t, that means

$$I_s \leq I_t.$$

The definition of I_t leads us to interpret $-\ln I_t$ as the losses in outstanding payments incurred during the period $(0, t]$ due to defaults. Thus we can interpret $-\Delta \ln I_s$, defined as

$$-\Delta \ln I_s := \ln I_t - \ln I_s \geq 0, \tag{2}$$

being the absolute value of losses incurred due to defaults during the holding period $[t, s]$.

The portfolio's *yield to maturity* Y_t at time t is then defined as the solution to the equation

$$P_t = \sum_{n=1}^{N} e^{-Y_t(t_n - t)} I_t c_n, \tag{3}$$

where P_t is the corporate bond portfolio's market price at time t. Thus we can express the portfolio's market price as $P_t = P(t, Y_t, I_t)$ where the function P is defined by

$$P(t, Y_t, I_t) = \sum_{n=1}^{N} e^{-Y_t(t_n - t)} I_t c_n.$$

Over a holding period $[t, s]$ where there are no changes made to the portfolio's composition, the log-return on the corporate bond portfolio can be defined by

$$\Delta \ln P_t := \ln P(s, Y_s, I_s) - \ln P(t, Y_t, I_t).$$

As derived in Koivu and Pennanen (2014), applying second order Taylor-approximation to $\ln P(s, Y_s, I_s)$ with respect to (s, Y_s, I_s) gives

$$\Delta \ln P_t \approx Y_s \Delta t - D_t \Delta Y_t + \Delta \ln I_t + \frac{1}{2}(C_t - D_t^2)\Delta Y_t^2. \tag{4}$$

where $\Delta t = s - t, \Delta Y_t = Y_s - Y_t, \Delta \ln I_t = \ln I_s - \ln I_t$ and

$$D_t = \frac{1}{P_t} \sum_{n=1}^{N} (t_n - t) e^{-Y_t(t_n - t)} I_t c_n$$

is the Macaulay Duration of the portfolio at time t, and

$$C_t = \frac{1}{P_t} \sum_{n=1}^{N} (t_n - t)^2 e^{-Y_t(t_n - t)} I_t c_n$$

is the Macaulay Convexity of the portfolio at time t. The term $Y_s \Delta t$ captures effects of time changes in log-returns; the term $D_t \Delta Y_t$ captures first-order effects of yield changes; the term $\frac{1}{2}(C_t - D_t^2)\Delta Y_t^2$

captures the second order effects of yield changes; and the term $\Delta \ln I_t$ is the return component arising from changes in outstanding payments due to default risk both systematic and idiosyncratic over $[t, s]$. Thus we interpret $-\Delta \ln I_t$ as losses in returns (default losses) due to both systematic and idiosyncratic default risks.

Equation (4) has the exact form as Koivu and Pennanen (2014)'s general two-factor return models before the underlying index I was interpreted economically for individual bond class. Removing the second order term gives us the first order approximation of the log-return

$$\Delta \ln P_t \approx Y_s \Delta t - D_t \Delta Y_t + \Delta \ln I_t. \tag{5}$$

Using (5), returns of corporate bonds are expressed as a function of time, yield-to-maturity and default losses, where the first factor is yield-to-maturity Y_s and the second factor is default losses $-\Delta \ln I_t$. This two-factor return model captures effects of both interest rate risk and default risk factor in corporate bond returns largely via Y_s and $-\Delta \ln I_t$.

We choose (5) as our two-factor return model for corporate bonds instead of (4) for several reasons: the first reason is that sensible historical data for Macaulay Convexity isn't available for the portfolio we study; the second reason is that both Koivu and Pennanen (2010) and Koivu and Pennanen (2014) showed that compared to the first order term, the second order term contributes marginally in explaining return variation of portfolio for various bond classes.

Koivu and Pennanen (2010)'s initial two-factor return model for corporate bonds was

$$\Delta \ln P_t \approx Y_s \Delta t - D_t \Delta Y_t + K_s, \tag{6}$$

where the first risk factor is yield-to-maturity Y_s and the second risk factor is K_s, which is also designed to capture effects of default risk on corporate bond returns. By comparing (5) and (6), it seems as if

$$K_s \approx \Delta \ln I_t. \tag{7}$$

This is in fact not true. That is simply because in Koivu and Pennanen (2010) the second risk factor K_s is used to approximate effects of defaults in returns caused by only the systematic default risk. This approximation was based on the assumption that the portfolio becomes well-diversified when it becomes infinitely large. Meanwhile default losses $-\Delta \ln I_t$ in our return model considers losses in returns of corporate bonds due to both systematic and idiosyncratic default risks. Koivu and Pennanen (2010) further assumed that there exists a risk-neutral measure under which the market prices of traded securities are equal to the expectations of their discounted cashflows. Under this assumption, the second risk factor K_s was approximated by a short-maturity yield spreads S_s between corporate bonds and government bonds,

$$K_s \approx -S_s \Delta t. \tag{8}$$

This approximation was validated in Koivu and Pennanen (2010) by comparing historical yield spreads and return residuals of a Merrill-Lynch investment grade corporate bond bond portfolios. The return residuals were computed as

$$\Delta \ln P_t - Y_s \Delta t + D_t \Delta Y_t. \tag{9}$$

We observe that yield spreads do have similar shape and magnitude of historical return residuals for the high yield corporate bonds in our empirical study, results of which are shown in Figure 1.

Substituting (8) into (6) gives Koivu and Pennanen (2010)'s two-factor return model for corporate bonds,

$$\Delta \ln P_t \approx (Y_s - \bar{a} S_s) \Delta t - D_t \Delta Y_t. \tag{10}$$

where $\bar{a}$ is a constant which can deviate from one since S_s is likely to underestimate/overestimate the average default loss rate in returns.

Instead of applying approximation to our second risk factor $-\Delta \ln I_t$ like Koivu and Pennanen (2010), we model $-\Delta \ln I_t$ stochastically considering both systematic and idiosyncratic risk. Our approach in modelling $-\Delta \ln I_t$ is largely motivated by results of empirical study in Figure 1 and Figure 2. This Figure shows that historical number of defaulted US issuers[4] has a very similar profile to the historical $-\Delta \ln I_t$ of a Bank of America Merrill Lynch's high yield corporate bond index. Using our default losses model, the two-factor return model (5) is not only suitable for the well-diversified corporate bond portfolios, but also the non well-diversified ones.

[4]Historical default issuers are collected from S&P's "Annual global corporate default study and rating transitions" studies. Please see section 5.1 for detailed explanation on data used to compute this Figure.

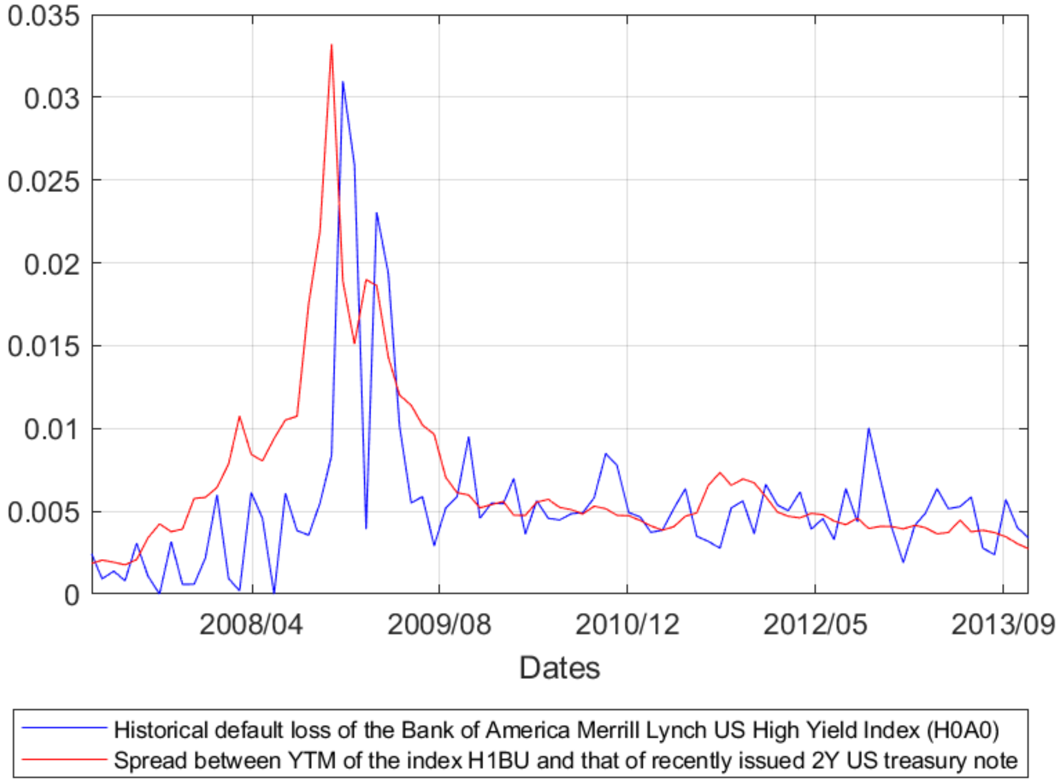

Figure 1: Comparison of $-\Delta \ln I_t$ and $S_s \Delta t$. This comparison shows that historical yield spreads have similar shape and magnitude of return residuals of the Bank of America Merrill Lynch US High Yield Index H0A0 from 31/01/1997 to 31/05/2014).

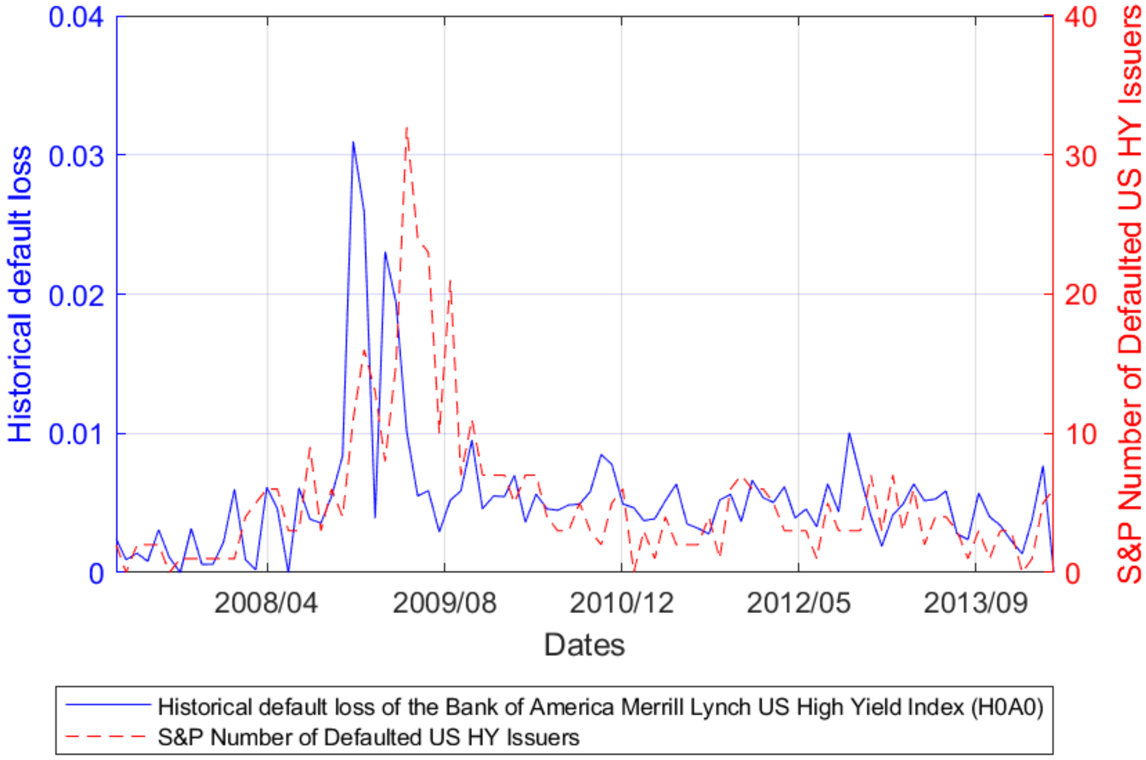

Figure 2: Comparison of $-\Delta \ln I_t$ (monthly historical data) and number of defaults (historical S&P US High Yield default data). This comparison shows that historical number of defaults have similar shape as the historical $-\Delta \ln I_t$. This empirical study was one of our motivations for our stochastic model of $-\Delta \ln I_t$.

4 A stochastic model for default losses

4.1 Modelling default losses using the compound (inhomogeneous) Poisson Gamma distribution

In this section we present our stochastic model for $-\Delta \ln I_t$. We approximate default losses $-\Delta \ln I_t$ in returns on the corporate bond portfolio by average default losses in returns on individual issuer. This approximation leads us to model $-\Delta \ln I_t$ over time as a compound (inhomogeneous) Poisson gamma process.

Consider an equally weighted M-issuer corporate bond portfolio in which each issuer is subject to default risk. Losses in returns on this bond portfolio incur when defaults happen. In an equally weighted M-issuer corporate bond portfolio, the contractual amount of mth issuer's nth outstanding payment denoted by c_n^m is Mth of the contractual amount of the portfolio's nth outstanding payments, that is

$$c_n^m = \frac{1}{M} c_n, \qquad n \in \{1, 2, \ldots, N\}, m \in \{1, 2, \ldots, M\} \tag{11}$$

We assume that the realised amount of mth issuer's nth outstanding payment at time t denoted by $c_{t,n}^m$ is reduced to $I_t^m c_n^m$ given its own defaults, that is

$$c_{t,n}^m = I_t^m c_n^m, \tag{12}$$

where $I_t^m \in [0, 1]$ is the recovery index representing remaining fraction of issuers m's outstanding payments due to its own defaults up to time t. Then using (1), (11) and (12), the amount of the portfolio's nth outstanding payment at time t can be expressed as

$$c_{t,n} = \sum_{m=1}^{M} c_{t,n}^m = \sum_{m=1}^{M} I_t^m c_n^m = \frac{1}{M} \sum_{m=1}^{M} I_t^m c_n = I_t c_n,$$

which gives

$$I_t = \frac{1}{M} \sum_{m=1}^{M} I_t^m. \tag{13}$$

The arithmetic mean of individual issuers' recovery indices can be approximated by their geometric mean using the first order Taylor-approximation,

$$\frac{1}{M} \sum_{m=1}^{M} I_t^m - 1 = \frac{1}{M} \sum_{m=1}^{M} (I_t^m - 1) \approx \frac{1}{M} \sum_{m=1}^{M} \ln I_t^m = \ln \prod_{m=1}^{M} (I_t^m)^{1/M} \approx \prod_{m=1}^{M} (I_t^m)^{1/M} - 1,$$

that is

$$\frac{1}{M} \sum_{m=1}^{M} I_t^m \approx \prod_{m=1}^{M} (I_t^m)^{1/M}.$$

Substituting the above approximation into equation (13) gives

$$I_t \approx \prod_{m=1}^{M} (I_t^m)^{1/M}. \tag{14}$$

Taking the log of both sides gives

$$-\ln I_t \approx \frac{1}{M} \sum_{m=1}^{M} -\ln I_t^m. \tag{15}$$

and their increments over $[t, s]$ are

$$-\Delta \ln I_t \approx \frac{1}{M} \sum_{m=1}^{M} (-\Delta \ln I_t^m). \tag{16}$$

If we assume that $(-\ln I_t^m)_{t\geq 0}$ is a compound Poisson gamma process and independent for each $m \in \{1,\ldots,M\}$, then $(-\ln I_t)_{t\geq 0}$ is a compound Poisson gamma process. Thus increments $-\Delta \ln I_t$ in (15) has a compound Poisson gamma distribution. Similar to Koivu and Pennanen (2010), the assumption that $(-\ln I_t^m)_{t\geq 0}$ is a compound Poisson gamma process can be justified by the *multiple defaults* approach in Schönbucher (1998) and Schönbucher (2003). Using this approach, each issuer is allowed to default multiple times. If we assume the number of defaults follow a Poisson process, and each loss given default is i.i.d with a gamma distribution, then losses on returns of each issuer $(-\ln I_t^m)_{t\geq 0}$ is a compound Poisson gamma process.

Proposition 4.1. If we assume the following

(i) Conditional on λ, the processes $(-\ln I^m, m = 1,\ldots,M)$ are independent and each $(-\ln I_t^m)_{t\geq 0}$ is a compound Poisson gamma process;

(ii) Conditional on λ, for any $s > t > 0$, increments $((-\Delta \ln I_t^m), m = 1,\ldots,M)$ are i.i.d with compound (inhomogeneous) Poisson gamma distribution $CPG(\lambda_s(s-t),\alpha_s,\beta_s)$ where α_s is the shape parameter and β_s is the inverse scale parameter of the gamma distribution.

Then according to (16), increments $-\Delta \ln I_t$ are independent and have a compound (inhomogeneous) Poisson gamma distribution $CPG(M\lambda_s(s-t),\alpha_s,M\beta_s)$; and according to (15), conditional on λ, the process $(-\ln I_t, t\geq 0)$ is a compound (inhomogeneous) Poisson process.

Proof. Due to the scaling properties of the gamma distribution, $-\frac{1}{M}\Delta \ln I_t^m$ for $m \in (1,\ldots,M)$ has a compound (inhomogeneous) Poisson gamma distribution $CPG(\lambda_s(s-t),\alpha_s,M\beta_s)$. Because sum of i.i.d random variables with a compound (inhomogeneous) Poisson gamma distribution also has a compound (inhomogeneous) Poisson gamma distribution, the increment $-\Delta \ln I_t$ as given by (16) has a compound (inhomogeneous) Poisson gamma distribution $CPG(M\lambda_s(s-t),\alpha_s,M\beta_s)$. Similarly, because sum of independent compound (inhomogeneous) Poisson gamma process is also a compound (inhomogeneous) Poisson gamma process, then according to (15), we have that $(-\ln I_t)_{t\geq 0}$ is also a compound (inhomogeneous) Poisson gamma process. □

Corollary 4.1. Conditional on λ, default losses $-\Delta \ln I_t$ converges in mean square to its mean $\mathrm{E}[-\Delta \ln \mathrm{I_t}]$ when M approaches infinity,

Proof. Conditional on λ, we have that

$$\mathrm{E}[(-\Delta \ln I_t - \mathrm{E}[-\Delta \ln I_t])^2] = \mathrm{E}[(-\Delta \ln I_t)^2] - \mathrm{E}[-\Delta \ln I_t]^2$$

Because $-\Delta \ln I_t$ has a compound Poisson gamma distribution $CPG(M\lambda_s(s-t),\alpha_s,M\beta_s)$, it can be expressed as

$$-\Delta \ln I_t = \sum_{k>N_t}^{N_s} x_k \tag{17}$$

where N_t denotes number of defaults happen up to time t and has a Poisson distribution $Pois(M\lambda_s(s-t))$, and $(x_k, k \in (N_t, N_s])$ are i.i.d random variables with a gamma distribution $G(\alpha_s, M\beta_s)$; furthermore $(x_k, k \in (N_t, N_s])$ are independent of N_t. Then we have

$$\begin{aligned}\mathrm{E}[(-\Delta \ln I_t)^2] &= \sum_{k=0}^{\infty} \mathrm{E}[(x_1 + x_2 + \cdots + x_k)^2 | N_s - N_t = k]\mathrm{P}(N_s - N_t = k)\\ &= \sum_{k=0}^{\infty} \left[k\mathrm{E}[x^2] + k(k-1)\mathrm{E}^2[x]\right]\mathrm{P}(N_s - N_t = k)\end{aligned}$$

where random variables $(x_i, i \in 1,\ldots,k)$ are i.i.d with gamma distribution $g(\alpha_s, M\beta_s)$. Using properties of both Poisson distribution and gamma distribution gives

$$\mathrm{E}[(-\Delta \ln I_t)^2] = \frac{\alpha_s(\alpha_s+1)}{M\beta_s^2}\lambda_s(s-t) + \left[\frac{\alpha_s}{\beta_s}\lambda_s(s-t)\right]^2 \tag{18}$$

Similary, we have

$$
\begin{aligned}
\mathrm{E}[-\Delta \ln I_t] &= \sum_{k=0}^{\infty} \mathrm{E}[x_1 + x_2 + \cdots + x_k | N_s - N_t = k] \mathrm{P}(N_s - N_t = k) \\
&= \lambda_s (s-t) \frac{\alpha_s}{\beta_s}. \qquad (19)
\end{aligned}
$$

Thus

$$
\mathrm{E}(-\Delta \ln I_t - \mathrm{E}[-\Delta \ln I_t])^2 = \frac{\alpha_s(\alpha_s + 1)}{M\beta_s^2} \lambda_s (s-t),
$$

from which we see that when $M \to \infty$, the variance $\mathrm{E}(-\Delta \ln I_t - \mathbb{E}[-\Delta \ln I_t])^2 = 0$. That is $-\Delta \ln I_t$ converges in mean square to $\mathbb{E}[-\Delta \ln I_t]$. □

4.2 Model parameter specification

We parametrise λ and β in $CPG(M\lambda_s(s-t), \alpha_s, M\beta_s)$ with historical short-term credit spreads s_s which are defined as differences between yield to maturity of high yield corporate bonds and government bonds. While λ represents default frequency and β controls sizes of losses, credit spreads are good indicators for both. Because when credit spreads increase, it is more likely to have more defaults and larger default losses. Furthermore it's shown in Koivu and Pennanen (2010)'s empirical study of investment grade bonds and our empirical study of high yield bonds (shown in Figure 1) that $S_s \Delta t$ captures evolution of $-\Delta \ln I_t$ very well.

We use the following two functional forms for λ_t and β_t,

Specifications	λ_t
Spec. 1	$\lambda_t = aS_t + b$
Spec. 2	$\lambda_t = \exp(a \ln S_t + b)$

(a) λ functional forms

Specifications	β_t
Spec. 1	$\beta_t = d$
Spec. 2	$\beta_t = c/\sqrt{S_t} + d$
Spec. 3	$\beta_t = c/S_t + d$
Spec. 4	$\beta_t = c/(S_t^{1.5}) + d$

(b) β functional forms

.

Table 1: Functional forms for both λ and β

Then for $CPG(M\lambda_s(s-t), \alpha_s, M\beta_s)$, there will be in total eight model specifications using combinations in table 1a and table 1b. Conditional on λ, the mean of $-\Delta \ln I_t$ is approximately $S_s \Delta t$ using any of those combinations, for example using combination of $\lambda_t = aS_t + b$ and $\beta_t = d$,

$$
\begin{aligned}
\mathrm{E}[-\Delta \ln I_t] &= \sum_{k=0}^{\infty} \mathrm{E}[x_1 + x_2 + \cdots + x_k | N_s - N_t = k] \mathrm{P}(N_s - N_t = k) \\
&= \sum_{k=0}^{\infty} e^{-\lambda_s(s-t)} \frac{(\lambda_s(s-t))^k}{k!} \frac{k\alpha_s}{\beta_s} \\
&= \lambda_s (s-t) \frac{\alpha_t}{\beta_s} \qquad (20) \\
&= (aS_s + b) \frac{\alpha}{d} \Delta t \qquad (21) \\
&\approx \bar{a} S_s \Delta t,
\end{aligned}
$$

where $\bar{a}$ is a constant and allowed to deviate from one.This recovers the approximation (8) in Koivu and Pennanen (2010). Specifically $\mathbb{E}[-\Delta \ln I_t]$ is closest to $S_s \Delta t$ using this combination, therefore it is not surprisingly to see later that simulated $-\Delta \ln I_t$ using combination of $\lambda_t = aS_t + b$ and $\beta_t = d$ best replicates historical $-\Delta \ln I_t$ as shown in section 6.

5 Parameter estimation

We discuss our approaches in estimating parameters of our default losses models in this section. We first show that estimating parameters of the compound Poisson gamma distribution $CPM(M\lambda_s(s-t), \alpha, M\beta_s)$ is equivalent to separately estimating parameters of a Poisson distribution $Pois(M\lambda_s(s-t))$ and a gamma distribution $G(k_t\alpha, M\beta_s)$ where k_t denotes all defaults happen in the portfolio during period $[t, s]$. We then discuss historical data required for parameter estimations of both $Pois(M\lambda_s(s-t))$ and $G(k_t\alpha, M\beta_s)$. Lastly we provide procedures for parameter estimations using the Maximum Likelihood method.

When monthly number of names M in the portfolio are known and denoted by M_t; the number of defaults (N_t, N_s) in (17) are known quantities denoted by (n_t, n_s), the equation (17) becomes

$$-\Delta \ln I_t = \sum_{k=1}^{k_t} x_k$$

where $k_t = n_s - n_t$. Given that $(x_k, k = 1, \ldots, k_t)$ is a sequence of i.i.d gamma random variables with gamma distribution $G(\alpha, M_t\beta_s)$, then $-\Delta \ln I_t$ has a gamma distribution $G(k_t\alpha, M_t\beta_s)$ because sum of i.i.d gamma random variables also has a gamma distribution. Thus when historical number of defaults are known, estimating parameters of $CPM(M\lambda_s(s-t), \alpha, M\beta_s)$ for $-\Delta \ln I_t$ is equivalent to separately estimating parameters of $Pois(M\lambda_s(s-t))$ for N_t and $G(k_t\alpha, M_t\beta_s)$ for $-\Delta \ln I_t$.

5.1 Historical Data

We discuss historical data for parameter estimations of both $Pois(M\lambda_s(s-t))$ and $G(k_t\alpha, M_t\beta_s)$. The portfolio we study is the Bank of America Merrill Lynch US High Yield corporate bond portfolio (H0A0). This index contains high yield rated bonds issued by US corporates and satisfying certain selection criteria. The Bank of America Merrill Lynch published an extensive amount of data associated with this index, such as monthly number of names in the index M_t, Macaulay Duration D_t and yield to maturity Y_t. The availability of those historical data facilitates our computation of historical default losses $-\Delta \ln I_t$. We choose portfolios of high yield bonds, because they are more susceptible to default risk and their losses in returns due to defaults are more likely to be higher than investment grade bonds.

To estimate parameters of the Poisson distribution and the gamma distribution, we need historical monthly number of names m_t, historical short-maturity credit spreads s_t and historical number of defaults k_t. We list sources below for all historical data required:

- For m_t, we use historical monthly number of names of the Bank of America Merrill Lynch US High Yield corporate bond portfolio (H0A0).
- For s_t, we use historical monthly short-maturity yields spreads between historical yield to maturity of the Bank of America Merrill Lynch 1-3 Year Single-B US High Yield Index (H1BU) and the yield to maturity of the most recently issued 2 year US treasury bond; historical $s_t\Delta t$ are plotted in Figure 1.
- For k_t, we collect historical monthly defaulted US high yield issuers from S&P's "Annual global corporate default study and rating transitions". These studies publishes annually defaulted issuers both investment grade and high yield around the global. The historical number of defaults k_t we use for parameter estimation are plotted in Figure 2.
- The last piece of historical data required to estimate parameters of the gamma distribution is the historical default losses $-\Delta \ln I_t$. We compute historical $-\Delta \ln I_t$ according to (9) using historical yield-to-maturity and duration published by the Bank of America Merrill Lynch US High Yield corporate bond portfolio (H0A0). The computed historical $-\Delta \ln I_t$ is displayed in both Figure 1 and Figure 2.

All historical data used for parameter estimation is from 31/01/2007 and 31/05/2014, for the reason that this is the period during which both historical credit spreads and historical number of defaults are available.

We study the above historical data. We observe that there exists a five months difference between peaks of the historical S&P number of defaults and the historical default losses $-\Delta \ln I_t$, as shown in Figure 2. When the historical S&P number of defaults is shifted five months backward in time, its peak coincides with that of $-\Delta \ln I_t$, as shown in Figure 3. Thus in estimating parameters of both λ_s and β_s, we use historical number of defaults reported at time $t+l$, where $l > 0$ denotes the reporting lag in months between the historical number of defaults and the historical default losses $-\Delta \ln I_t$. Using the reporting lag l is justified by the fact that defaults usually happens before they are recorded[5]. Differences between actual default dates and recorded default dates are also observed in an empirical study by Guo, Jarrow, and Lin (2008). The bond sample set they used covered virtually the entire U.S. and international bond market from 1984 to 2007. Their study showed that estimated actual default dates of chosen sample bonds are ahead of recorded default dates. More specifically, biggest difference in time between both dates was 180 days. This is similar to the five months gaps observed in our empirical study, as shown in Figure 2.

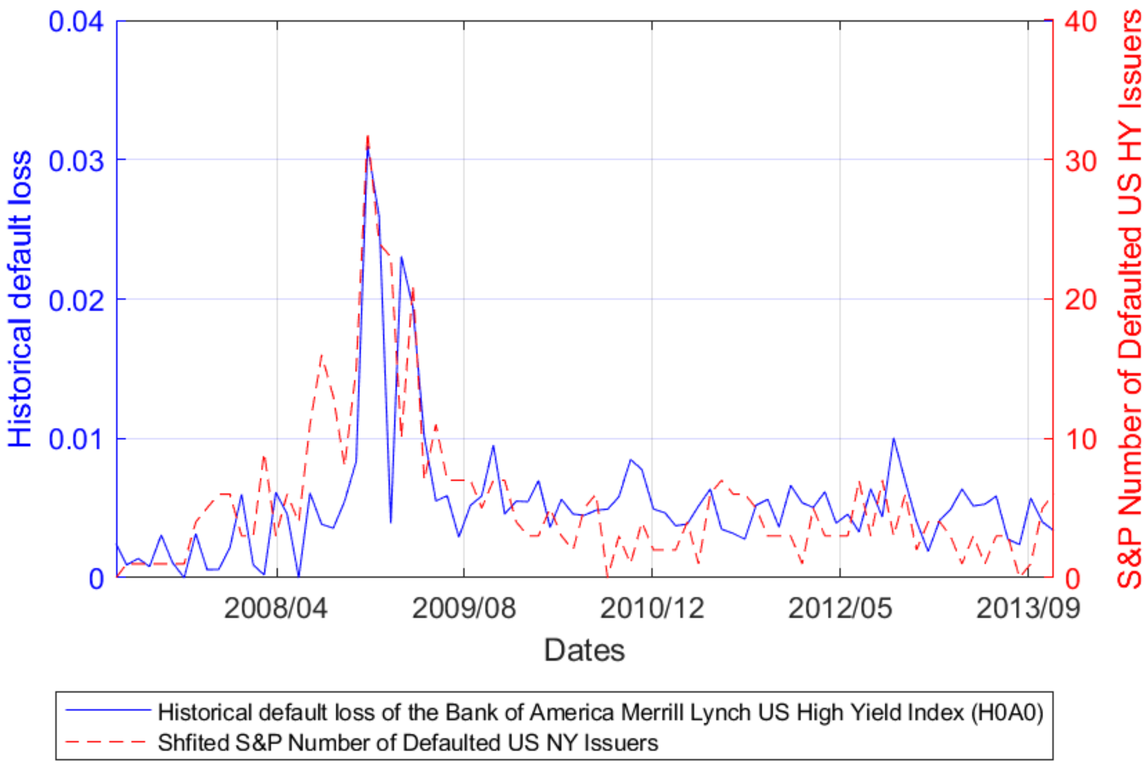

Figure 3: The peak of historical S&P number of US HY defaults data reported in five months coincides with the peak of the historical default losses of return $-\Delta \ln I_t$ of the Bank of America Merrill Lynch US High Yield Index (H0A0) from 31/01/2007 to 31/05/2014.

5.2 Maximum likelihood estimate for the Inhomogeneous Poisson distribution

To estimate parameters of an inhomogeneous Poisson distribution $Pois(M\lambda_s(s-t))$ with the Maximum Likelihood method, firstly we partition the observation period of the given set of historical data denoted by $[0, T]$ into n equally spaced subintervals. Each endpoint $0 = t_0 < t_1 < t_2 < \cdots < t_n = T$ denotes the beginning of a month. A month is thus defined by the subinterval starting at t_{n-1} and ending at t_n, during which there is no change to the portfolio composition.

Let $(N_t)_{t\geq 0}$ denote the inhomogeneous Poisson Process whose increments $\Delta N_{t_1} := N_{t_1} - N_{t_0}, \dots,$ $\Delta N_{t_n} := N_{t_n} - N_{t_{n-1s}}$ are independent random variables with inhomogeneous Poisson distributions

[5] By S&P's definition, defaults are assumed to take place on the earliest of: the date S&P's revised the rating to 'D'; the date a debt payment was missed; the date a distressed exchange offer was announced or the date the debtor filed for, or was forced into, bankruptcy. By S&P's definition, Distressed exchanges are considered defaults whenever the debt holders are coerced into accepting substitute instruments with lower coupons, longer maturities, or any other diminished financial terms.

$Pois(M_{t_1}\lambda_{t_1}(t_1-t_0))$, ..., $Pois(M_{t_n}\lambda_{t_n}(t_n-t_{n-1}))$, where M_{t_n} denotes the monthly number of issuers in the index H0A0 over $[t_{n-1},t_n]$. The probability density $\mathrm{p}(\Delta N_t = k_t)$ that monthly number of defaults ΔN_t equals to k_t is defined by

$$\mathrm{p}(\Delta N_t = k_t) = e^{-M_t\lambda_s(s-t)}\frac{(M_t\lambda_s(s-t))^{k_t}}{k_t!}.$$

Thus conditional on λ, the log-likelihood function is

$$\sum_{i=1}^{n}\ln \mathrm{p}(\Delta N_{t_i} = k_{t_i}) = -\sum_{i=1}^{n} M_{t_i}\lambda_{t_{i+1}}(t_{i+1}-t_i) + \sum_{i=1}^{n} k_{t_i}\ln[M_{t_i}\lambda_{t_{i+1}}(t_{i+1}-t_i)] - \sum_{i=1}^{n}\ln k_{t_i}!,$$

in which $\lambda_{t_{i+1}}$ is parametrised using functional forms in Table 1a. Parameters of those functional forms are found by maximising the log-likelihood function (or minimising the negative of the log-likelihood function) using function **fmincon** in MATLAB. The function **fmincon** is used to find minimum of constrained nonlinear multivariate function. The Matlab code used for this purpose is included in the Appendix A.

5.2.1 Standard Errors

In order to assess parameter robustness, we compute standard errors of parameters estimated using the maximum likelihood method for all λ functional forms. We first derive standard errors of parameters for each λ functional form in this section, then we present corresponding numerical results in section 6.1.1.

- $\lambda_t = aS_t + b$.

  The log-likelihood function now becomes

  $$\ln L = -\sum_{i=1}^{n} M_{t_i}(aS_{t_{i+1}}+b)(t_{i+1}-t_i) + \sum_{i=1}^{n} k_{t_i}\ln[M_{t_i}(aS_{t_{i+1}}+b)(t_{i+1}-t_i)] - \sum_{i=1}^{n}\ln k_{t_i}!,$$

  we have the following second order partial derivatives

  $$\frac{\partial^2 \ln L}{\partial a^2} = -\sum_{i=1}^{n} k_i\frac{S_{t_{i+1}}^2}{(aS_{t_{i+1}}+b)^2} \quad \text{and} \quad \frac{\partial^2 \ln L}{\partial b^2} = -\sum_{i=1}^{n}\frac{k_i}{(aS_{t_{i+1}}+b)^2}.$$

- $\lambda_t = \exp(a\ln S_t + b)$.

  The log-likelihood function now becomes

  $$\ln L = -\sum_{i=1}^{n} M_{t_i}\exp(a\ln S_{t_{i+1}}+b)(t_{i+1}-t_i) + \sum_{i=1}^{n} k_{t_i}\ln[M_{t_i}\exp(a\ln S_{t_{i+1}}+b)(t_{i+1}-t_i)] - \sum_{i=1}^{n}\ln k_{t_i}!,$$

  we have the following second order partial derivatives

  $$\frac{\partial^2 \ln L}{\partial a^2} = -\sum_{i=1}^{n} M_i\exp(a\ln S_{t_{i+1}}+b)(t_{i+1}-t_i)(\ln S_{t_{i+1}})^2,$$
  $$\frac{\partial^2 \ln L}{\partial b^2} = -\sum_{i=1}^{n} M_i\exp(a\ln S_{t_{i+1}}+b)(t_{i+1}-t_i).$$

So the standard error of a denoted by S_a and the standard error of b denoted by S_b are respectively defined as

$$S_a = \sqrt{-\mathrm{E}[\frac{\partial^2 \ln L}{\partial a^2}]^{-1}/n} \quad \text{and} \quad S_b = \sqrt{-\mathrm{E}[\frac{\partial^2 \ln L}{\partial b^2}]^{-1}/n}.$$

5.3 Maximum likelihood estimate for the gamma distribution

Parameters of a gamma distribution $G(k_t\alpha, M_t\beta_s)$ are similarly estimated using the Maximum Likelihood method. We use the same observation period defined in the previous subsection 5.2.

When historical number of defaults are known, historical default losses $-\Delta\ln I_t$ has a gamma distribution $G(k_t\alpha, M_t\beta_s)$. The probability density $\mathrm{p}(-\Delta\ln I_t = l_t)$ that $-\Delta\ln I_t$ takes historical monthly default losses denoted by l_t at time t is defined by

$$\mathrm{p}(-\Delta\ln I_t = l_t) = \frac{(M_t\beta_s)^{k_t\alpha}}{\Gamma(k_t\alpha)} l_t^{k_t\alpha-1} e^{-M_t\beta_s l_t}.$$

Then the log-likelihood function over the observation period given α and $\beta_{t_1},\ldots,\beta_{t_n}$ is

$$\sum_{i=1}^{n}\ln\mathrm{p}(-\Delta\ln I_{t_i} = l_{t_i}) = \alpha\sum_{i=1}^{n}k_{t_i}\ln(M_{t_i}\beta_{t_{i+1}}) + \sum_{i=1}^{n}(k_{t_i}\alpha-1)\ln l_{t_i} - \sum_{i=1}^{n}M_{t_i}\beta_{t_{i+1}}l_{t_i} - \sum_{i=1}^{n}\ln\Gamma(k_{t_i}\alpha),$$

in which parameter $\beta_{t_{i+1}}$ are parametrised using functional forms in section 4. Parameters of functional forms used to describe $\beta_{t_{i+1}}$ are found by maximising the log-likelihood function (or minimising the negative of the log-likelihood function) using function **fmincon** in MATLAB. The Matlab code used for this purpose is included in the Appendix A.

5.3.1 Standard Errors

Similarly we derive standard errors of the maximum likelihood estimates of α and parameters used in all β functional forms in this section; subsequently we present corresponding numerical results in section 6.1.1.

- $\beta_t = d$.

  The log-likelihood function now becomes

  $$\ln L = \alpha\sum_{i=1}^{n}k_{t_i}\ln(M_{t_i}d) + \sum_{i=1}^{n}(k_{t_i}\alpha-1)\ln l_{t_i} - \sum_{i=1}^{n}M_{t_i}l_{t_i}d - \sum_{i=1}^{n}\ln\Gamma(k_{t_i}\alpha),$$

  we have the following second order partial derivatives

  $$\frac{\partial^2\ln L}{\partial d^2} = -\frac{\alpha}{d^2}\sum_{i=1}^{n}k_{t_i} \quad\text{and}\quad \frac{\partial^2\ln L}{\partial\alpha^2} = -\sum_{i=1}^{n}\left(\frac{\Gamma'(k_{t_i}\alpha)}{\Gamma(k_{t_i}\alpha)}\right)'.$$

- $\beta_t = c/S_t^a + d$, where $a = 0.5, 1, 1.5$.

  The log-likelihood function now becomes

  $$\ln L = \alpha\sum_{i=1}^{n}k_{t_i}\ln(M_{t_i}(c/S_{t_{i+1}}^a + d)) + \sum_{i=1}^{n}(k_{t_i}\alpha-1)\ln l_{t_i} - \sum_{i=1}^{n}M_{t_i}l_{t_i}(c/S_{t_{i+1}}^a + d) - \sum_{i=1}^{n}\ln\Gamma(k_{t_i}\alpha),$$

  we have the following second order partial derivatives

  $$\frac{\partial^2\ln L}{\partial\alpha^2} = -\sum_{i=1}^{n}\left(\frac{\Gamma'(k_{t_i}\alpha)}{\Gamma(k_{t_i}\alpha)}\right)', \quad \frac{\partial^2\ln L}{\partial d^2} = -\alpha\sum_{i=1}^{n}\frac{k_{t_i}}{(d+\frac{c}{S_{t_{i+1}}^a})^2}, \quad\text{and}$$
  $$\frac{\partial^2\ln L}{\partial c^2} = -\alpha\sum_{i=1}^{n}\frac{k_{t_i}}{(dS_{t_{i+1}}^a + c)^2}.$$

So the standard error of α denoted by S_α, the standard error of c denoted by S_c and the standard error of d denoted by S_d are respectively defined as

$$S_\alpha = \sqrt{-\mathrm{E}[\frac{\partial^2\ln L}{\partial\alpha^2}]^{-1}/n}, \quad S_d = \sqrt{-\mathrm{E}[\frac{\partial^2\ln L}{\partial d^2}]^{-1}/n} \quad\text{and}\quad S_c = \sqrt{-\mathrm{E}[\frac{\partial^2\ln L}{\partial c^2}]^{-1}/n}.$$

6 Numerical results

In this section, we present the compound Poisson gamma model specification which produces reasonable small values of information criteria and simulated $-\Delta \ln I_t$ most resembling historical $-\Delta \ln I_t$. We also display the simulated $-\Delta \ln I_t$ with the chosen compound Poisson gamma model specification; the simulated number of defaults N_t using the Poisson distribution with the same λ functional form; and the simulated $-\Delta \ln I_t$ using the gamma distribution with the same β functional form.

The information criteria we chose for model selection is AICc/AICc difference. Information criteria AIC stands for Akaike's Information Criterion and is defined by (Burnham and Anderson (2002))

$$AIC = -2\ln(L) + 2k,$$

where L denotes the maximum value of the likelihood function and k denotes the total number of estimated parameters. AICc is AIC with a correction for small sample sizes and defined by

$$AICc = AIC + \frac{2k^2 + 2k}{n - k - 1},$$

where n denotes the sample size. The model with the smallest value of AICc is the one that is estimated to be "closest" to the unknown reality that generated the data from among the candidate models considered. Information criteria $AICc$ is recommended over AIC for model selection when the sample size is small relative to the number of parameters. In general, it was advised in Burnham and Anderson (2002) to use $AICc$ when the ratio n/k is small (say < 40). Given the sample size of our dataset is 86 and the number of parameters for estimations is bigger or equivalent to 3, we thus choose $AICc$ over AIC as the information criteria for our model selection.

However it is also noted in Burnham and Anderson (2002) that it is not that absolute size of the $AICc$ value matters but $AICc$ difference in model selection. Information criteria $AICc$ difference, denoted by $\Delta AICc$, is computed as

$$\Delta AICc = AICc - \min_{i \in \{1,\ldots,m\}} AICc_i,$$

assuming that there are in total m candidate models. When $\Delta AICc$ is between 0 to 2, it indicates that the level of empirical support for this candidate model is substantial; when $\Delta AICc$ is between 4 to 7, it indicates that there is considerable less level of empirical support for this candidate model; when $\Delta AICc$ is bigger than 10, it indicates that there is essentially no empirical support for this candidate model.

We don't use information criteria AICc/$\Delta AICc$ as the single criteria for model selection, because the model with the smallest AICc is not necessarily the best as discussed in Burnham and Anderson (2002). Details of the simulation algorithms and corresponding Matlab code is included in Appendix B.

6.1 Estimated parameters of the compound (inhomogeneous) Poisson gamma model

We compute Δ_{AICc} and simulate $-\Delta \ln I_t$ using all combinations of the λ function forms in table 1a, the β functional forms in table 1b and reporting lags from one to twelve months.

Among all computed Δ_{AICc} and simulated $-\Delta \ln I_t$, we find that the compound Poisson gamma model specification using combination of the λ functional form $\lambda_t = aS_t + b$, the β functional form $\beta_t = \beta$ and two months reporting lag produces a reasonably small value of Δ_{AICc} as shown in table 2. This combination also producess simulated $-\Delta \ln I_t$ which replicates well the historical default losses and will be displayed in section 6.2.

Lags	Parameters	$\beta_t = \beta$
Lag=2	$\lambda_t = aS_t + b$	-248.55804

(a) Value of AICc

Lags	Parameters	$\beta_t = \beta$
Lag=2	$\lambda_t = aS_t + b$	5.80952

(b) Value of Δ_{AICc}

.

Table 2: AICc and $\Delta AICc$ of the compound (inhomogeneous) Poisson gamma distribution $CPG(M_t\lambda_s(s-t), \alpha_t, M_t\beta_s)$ using estimated parameters of functional form $\lambda_t = aS_t + b$ and $\beta_t = \beta$ with lag=2 as shown in Table 4.

We then inspect $AICc$ and Δ_{AICc} of the Poisson distribution $Pois(M_t\lambda_s(s-t))$ using functional form $\lambda_t = aS_t + b$ and the gamma distribution $G(k_t\alpha, M_t\beta_s)$ using functional $\beta_t = \beta$. We see in table 4a that Δ_{AICc} of the Poisson distribution is large, which indicates that functional form $\lambda_t = aS_t + b$ is not a good candidate model for the inhomogeneous Poisson distribution $Pois(M_t\lambda_s(s-t))$. Despite this large Δ_{AICc}, later in the next section we show that simulated number of defaults using the Poisson distribution with this $\lambda_t = aS_t + b$ turns out to have a similar shape as the historical number of defaults well. We also see from table 4b that the Δ_{AICc} of the gamma distribution is reasonably small, which indicates that functional form $\beta_s = \beta$ is a suitable choice for the gamma distribution.

Lag	Sample Size	Parameters	$\lambda_s = aS_s + b$
Lag=2	86	a	0.48779
		b	-0.00043
		AICc	409.61443
		Δ_{AICc}	28.42571

(a) Parameters of functional form $\lambda_t = aS_t+b$ with lag=2 and corresponding AICc/$\Delta AICc$ of the inhomogeneous Poisson distribution $Pois(M_t\lambda_s(s-t))$.

Lags	Parameters	$\beta_s = \beta$
Lag=2	α	0.36871
	c (or β)	0.18956
	AICc	-658.37715
	Δ_{AICc}	3.917591

(b) Parameters α and β using functional form $\beta_s = \beta$ with lag=2, as well as corresponding AICc/$\Delta AICc$ of the gamma distribution $G(k_t\alpha, M_t\beta_s)$

Table 4: Estimated parameters of λ_s and β_s.

6.1.1 Standard errors

We numerically compute standard errors of estimated parameters of the compound (inhomogeneous) Poisson Gamma model, for combinations of all reporting lags, λ functional forms and β functional forms.

Table 5 shows standard errors of parameters in all λ functional forms, while table 6 shows standard errors of α and parameters in all β functional forms. From both tables, we see that standard errors of maximum likelihood estimates for each parameter, across all functional forms (either for λ or for β) and all reporting lags, are small and do not differ significantly. These results indicate that precisions of maximum likelihood estimates for parameters of all 96 models are similarly good; they also show that the estimated parameters of compound (inhomogeneous) Poisson Gamma models are robust.

	$\lambda_s = aS_s + b$		$\lambda_s = a \ln S_s + b$	
Lags(days)	a	b	a	b
0	0.00207	0.00001	0.00350	0.01721
1	0.00206	0.00001	0.00355	0.01725
2	0.00209	0.00001	0.00359	0.01734
3	0.00213	0.00001	0.00361	0.01748
4	0.00215	0.00001	0.00365	0.01762
5	0.00216	0.00001	0.00371	0.01778
6	0.00220	0.00001	0.00373	0.01788
7	0.00223	0.00001	0.00373	0.01800
8	0.00225	0.00001	0.00372	0.01812
9	0.00227	0.00001	0.00373	0.01825
10	0.00228	0.00001	0.00372	0.01839
11	0.00228	0.00002	0.00372	0.01851
12	0.00228	0.00002	0.00373	0.01867

Table 5: Standard errors of parameters of λ functional forms in table 1a

	$\beta_s = d$		$\beta_s = c/\sqrt{S_s} + d$			$\beta_s = c/S_s + d$			$\beta_s = c/S_s^{1.5} + d$		
Lags(days)	α	d	α	c	d	α	c	d	α	c	d
0	0.84362	0.00021	0.85967	0.00001	0.00019	0.86051	6.48565E-07	0.00019	0.85641	3.38995E-08	0.00020
1	0.91416	0.00024	0.94276	0.00001	0.00021	0.93995	6.20927E-07	0.00020	0.93311	2.99405E-08	0.00021
2	0.94863	0.00025	0.98759	0.00001	0.00022	0.98124	7.07824E-07	0.00022	0.97106	3.71891E-08	0.00023
3	0.84033	0.00023	0.86272	0.00001	0.00020	0.85998	6.50667E-07	0.00020	0.85410	3.41030E-08	0.00021
4	0.91662	0.00024	0.94893	0.00001	0.00021	0.94397	6.85962E-07	0.00021	0.93658	3.56920E-08	0.00022
5	0.84809	0.00023	0.87891	0.00001	0.00019	0.87440	4.90791E-07	0.00018	0.86812	2.24369E-08	0.00018
6	0.91468	0.00024	0.95226	0.00052	0.00648	0.94576	6.13709E-07	0.00020	0.93775	3.11747E-08	0.00021
7	0.81123	0.00021	0.82496	0.00001	0.00019	0.82285	6.52987E-07	0.00019	0.82011	3.34438E-08	0.00020
8	0.78489	0.00021	0.78944	0.00001	0.00020	0.78864	7.73897E-07	0.00020	0.71279	4.43219E-08	0.00040
9	0.70817	0.00019	0.70900	0.00001	0.00019	0.70871	7.95178E-07	0.00019	0.70898	4.58009E-08	0.00019
10	0.67717	0.00018	0.67735	0.00001	0.00018	0.67800	8.45633E-07	0.00018	0.67827	5.02138E-08	0.00019
11	0.70601	0.00019	0.70896	0.00001	0.00019	0.70934	9.46087E-07	0.00019	0.84715	1.19952E-07	0.00109
12	0.70302	0.00019	0.70904	0.00002	0.00020	0.70775	9.93381E-07	0.00020	0.70852	6.07420E-08	0.00019

Table 6: Standard errors of α and parameters of β functional forms in table 1b

6.2 Analysis of simulated $-\Delta \ln I_t$ VS. historical $-\Delta \ln I_t$

We simulate number of defaults with a Poisson distribution $Pois(M_t\lambda_s(s-t))$ using combinations of λ functional forms in table 1a and reporting lags ranging from one to twelve months; default losses with a gamma distribution $G(k_t\alpha, M_t\beta_s)$ using combinations of β functional forms in table 1b and reporting lags from one to twelve months; and default losses with a compound Poisson gamma distribution $CPG(M_t\lambda_s(s-t), \alpha_t, M_t\beta_s)$ using all combination of λ functional forms, and β functional forms as well as all reporting lags.

To simulate both Poisson and gamma distributions, historical monthly number of names, historical spreads and number of defaults are required. We use historical data from 31/01/2007 to 31/05/2014, because both historical dataset are available in this period. In addition, historical data of this period has been used for parameter estimations of both the Poisson and gamma distributions; using the same data in simulation is a good way to validate how well model parameters were estimated.

To simulate default losses using the compound (inhomogeneous) Poisson gamma distribution, only historical monthly number of names and historical spreads are required. In order to assess how well parameters of the model are estimated and how well specified the model is, we firstly use dataset from 31/01/1997 (the inception of the BoAML H0A0 index) until 31/05/2014, which covers both the period used for parameter estimation and the period before.

Among all simulated default losses using the compound (inhomogeneous) Poisson gamma distribution $CPG(M_t\lambda_s(s-t), \alpha, M_t\beta_s)$, we find that the combination of $\lambda_t = aS_t + b$ and $\beta_t = \beta$ and two months reporting lag produces simulated $-\Delta \ln I_t$ that resembles historical default losses from 31/01/1997 to 31/05/2014 very closely as shown in Figure 4. We observe from this figure that

- the historical $-\Delta \ln I_t$ sits within the 90th confidence band of the simulated $-\Delta \ln I_t$;
- the magnitude of the 50th percentile of the simulated $-\Delta \ln I_t$ is very close to that of the historical $-\Delta \ln I_t$; both have very similar profiles; 5.

Those observations combined with the small ΔAIC_c discussed in the previous section suggest that the compound Poisson gamma distribution with the combination of $\lambda_t = aS_t + b$ and $\beta_t = \beta$ and two months reporting lag is a good candidate model for $-\Delta \ln I_t$.

We then inspect whether the simulated Poisson distribution using $\lambda_t = aS_t + b$ and two months reporting lag, the simulated Gamma distribution using $\beta_t = \beta$ and two months reporting lag are reasonable representations of historical number of defaults and default losses respectively. From Figure 5, we see that the simulated number of defaults have a similar shape as that of the historical number of defaults; we also see that simulated number of defaults peaks four months ahead of the historical number of defaults instead of two months. That is because historical short-term credit spreads used in parametrising intensity λ_t peak at the same time as the historical default losses $-\Delta \ln I_t$ but five months earlier than the historical number of defaults.

From Figure 6, we see that the simulated $-\Delta \ln I_t$ using the Gamma distribution with $\beta_t = \beta$ and two months reporting lag is good given close resemblance between the simulated and the historical $-\Delta \ln I_t$. We can also see from this Figure that the simulated $-\Delta \ln I_t$ peaks three months later than the historical one but at the same time as the shifted historical number of defaults. That is because the shifted historical number of defaults are used directly in simulating the gamma distribution as discussed in section 6.1.

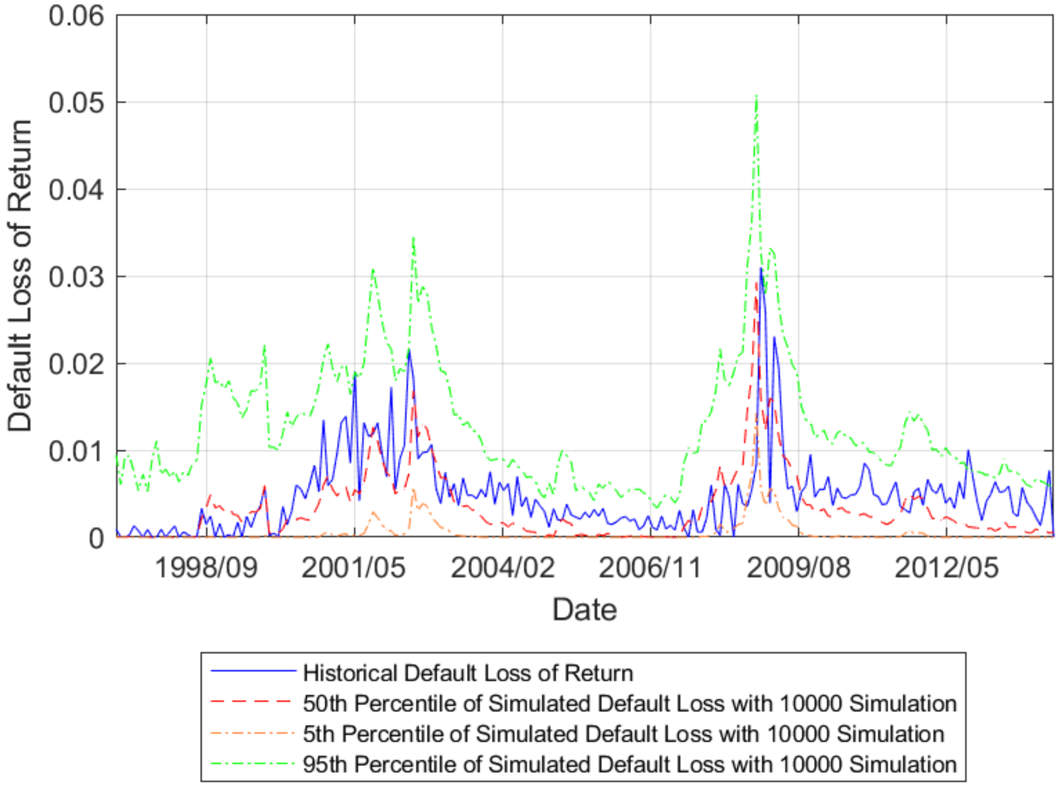

Figure 4: Simulated default losses $-\Delta \ln I_t$, produced with compound Poisson gamma distribution using the combination of $\lambda_t = aS_t + b$ and $\beta_t = \beta$ and two months reporting lag, closely resembles the historical default losses from 31/01/1997 to 31/05/2014.

We thus conclude that compound Poisson gamma distribution $CPG(M_t\lambda_s(s-t), \alpha_t, M_t\beta_s)$ with the combination of functional forms $\lambda_t = aS_t + b$ and $\beta_t = \beta$ and two months reporting lag is a good candidate model for $-\Delta \ln I_t$.

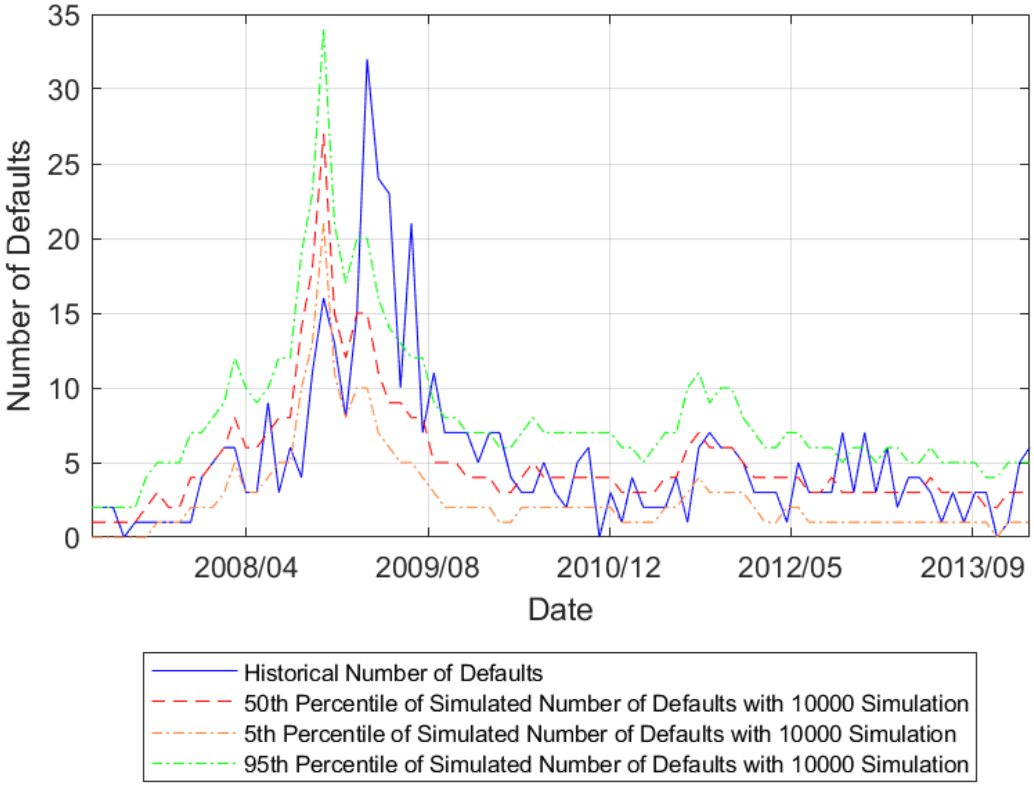

Figure 5: Simulated number of defaults, produced using a Poisson distribution using $\lambda_t = aS_t + b$ and two months reporting lag, have similar profiles as historical S&P number of defaults.

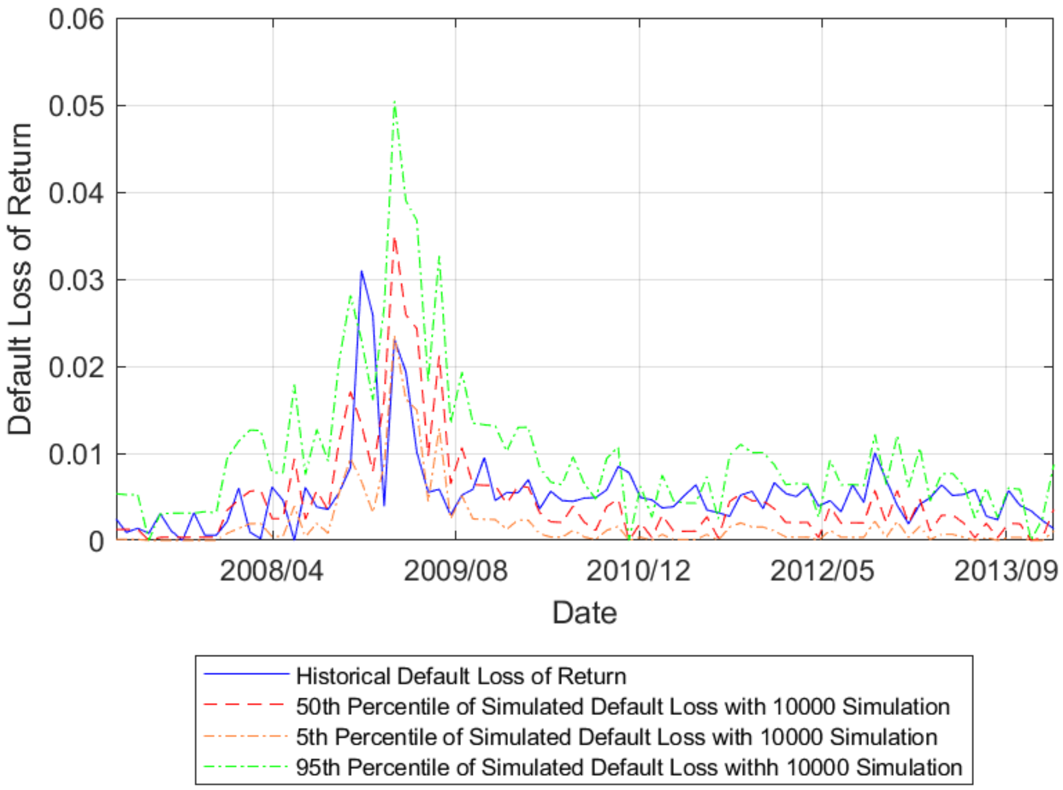

Figure 6: Simulated default losses $-\Delta \ln I_t$, which are produced using the gamma distribution with $\beta_t = \beta$ and two months reporting lag closely resembles historical default losses.

6.2.1 Performance of the selected default losses model

In order to assess the performance of our proposed model for default losses, we use additional historical dataset from 31/05/2014 until 31/12/2019 to simulate default losses. We then compare simulated default losses and historical ones to see how good the model is using parameters estimated with past data.

Figure 7 shows simulated $-\Delta \ln I_t$ using the selected default losses model and historical dataset from 31/01/1997 to 31/05/2014, as well as from 31/05/2014 to 31/12/2019. We plot simulated results from both periods together in one figure, such that we can see a direct comparison between model performance using the test dataset from 31/01/1997 to 31/05/2014, and that using the validation dataset from 31/05/2014 to 31/12/2019.

We observe in Figure 7 that simulated $-\Delta \ln I_t$ resembles historical default losses from 31/05/2014 to 31/12/2019 very closely, similar to observations made in the previous section for simulated results from 31/01/1997 to 31/05/2014. We observe that during the period from 31/05/2014 to 31/12/2019

- the historical $-\Delta \ln I_t$ sits within the 90th confidence band of the simulated $-\Delta \ln I_t$;
- the magnitude of the 50th percentile of the simulated $-\Delta \ln I_t$ is very close to that of the historical $-\Delta \ln I_t$; both have very similar profiles;

This observation gives us confidence that our default loss model has reasonably good performance.

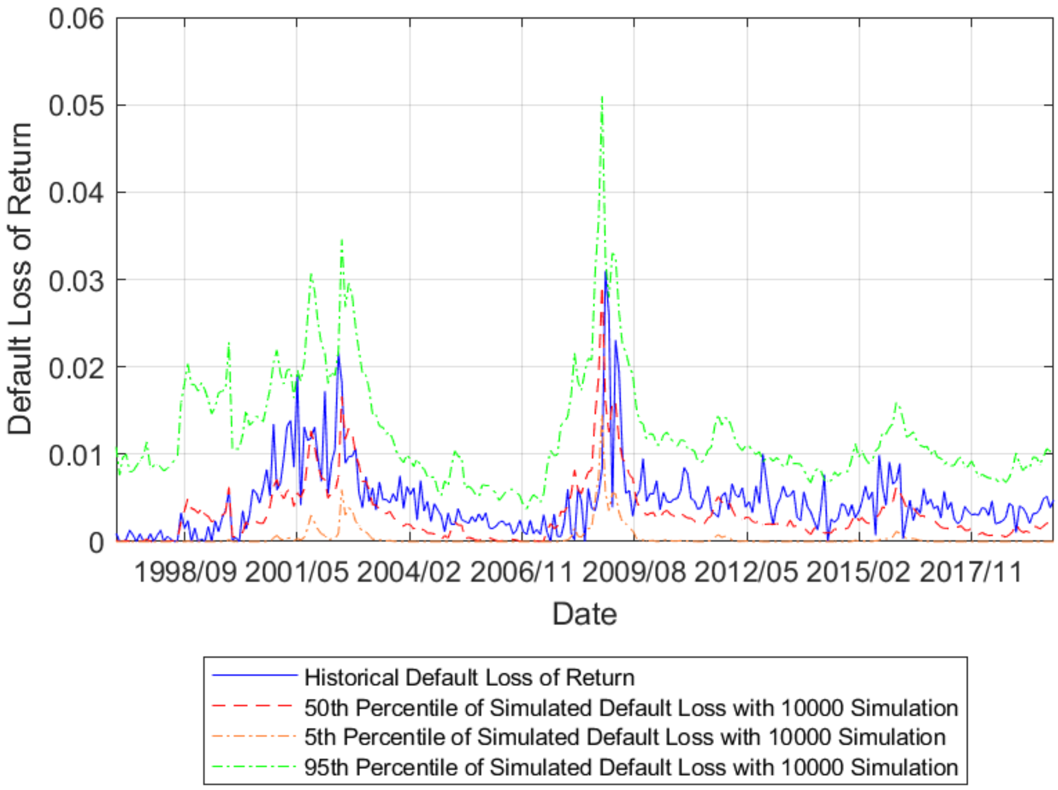

Figure 7: Simulated default losses $-\Delta \ln I_t$, produced with compound Poisson gamma distribution using the combination of $\lambda_t = aS_t + b$ and $\beta_t = \beta$ and two months reporting lag, closely resembles the historical default losses from 31/01/1997 to 31/12/2019.

7 Conclusion

We have come up with a stochastic model for default losses in returns on corporate bonds due to both systematic and idiosyncratic default risks in returns on corporate bonds. Using our default losses model, our two-factor return model for corporate bonds, based on Koivu and Pennanen (2014), is suitable for both well-diversified and non well-diversified corporate bond portfolios.

We use the underlying index in Koivu and Pennanen (2014) for corporate bonds to represent the remaining fraction of all outstanding payments due to both systematic and idiosyncratic default risks. Then returns of corporate bonds are expressed in terms of time, yield-to-maturity and default losses, which is a function of the underlying index. Effects of systematic and idiosyncratic default risks on returns are captured by the default losses component. We model the default losses over a portfolio holding period as a compound (inhomogeneous) Poisson gamma distribution. We choose this modelling approach is partly motivated by our empirical study of S&P's historical number of defaults and historical default losses of Bank of America Merill Lynch US High Yield corporate bond index; it's also because default losses in returns on corporate bonds are approximated as an average of default losses in returns on individual issuers, which in turn is assumed to have a compound (inhomogeneous) Poisson gamma distribution based on familiar mathematical justifications.

We parametrise the compound (inhomogeneous) Poisson gamma distribution and end up with 96 model specifications. For each model specification, we estimate model parameters and subsequently simulate $-\Delta \ln I_t$. Among all results, we find that the compound Poisson gamma distribution with $\lambda_t = aS_t + b$ and $\beta_t = \beta$ produces small enough information criteria as well as simulated $-\Delta \ln I_t$ that closely resembles historical $-\Delta \ln I_t$. We thus conclude that our default losses model is suitable for modelling default risk including both systematic and idiosyncratic risks in returns of corporate bonds.

With our proposed default losses model, this two-factor return model is particularly useful for portfolio analysis and risk management where dynamic statistical return models are required; it could potentially be useful for factor investing and bond index construction where contributions to bond returns from individual risk factors need to be known; finally it could also be useful for predicting future bond returns and corresponding strategy planning.

Appendix A Matlab code for parameter estimations

A.1 Parameter estimations for the inhomogeneous Poisson distribution

We use the following Matlab function to estimate parameters of the inhomogeneous Poisson distribution $Pois(M_t\lambda_s(s-t))$.

```
        year_selected = 'Master Year';
        time_period = 'monthly';
        with_cds_spread = 'false';

        %model configuration
        with_num_names = 'true';

        %Poisson model configuration
        modelchoice='linearb'; %linear, linearb, loglinear1, loglinear2=exp(b+alnS), linear2=aS^2,linearb2=aS^2+b - S^2 not good

        %shifting data configuration
        shifted_flag = 'true';
        shiftedDays = 2;

        if strcmp(time_period,'monthly')
        if strcmp(shifted_flag, 'true')
        if shiftedDays==1
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND1.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND1.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND1.xlsx'});
        elseif shiftedDays==2
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND2.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND2.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND2.xlsx'});
        elseif shiftedDays==3
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND3.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND3.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND3.xlsx'});
        elseif shiftedDays==4
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND4.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND4.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND4.xlsx'});
        elseif shiftedDays==5
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND.xlsx'});
```

```
    elseif shiftedDays==6
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND6.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND6.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND6.xlsx'});
    elseif shiftedDays==7
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND7.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND7.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND7.xlsx'});
    elseif shiftedDays==8
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND8.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND8.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND8.xlsx'});
    elseif shiftedDays==9
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND9.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND9.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND9.xlsx'});
    elseif shiftedDays==10
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND10.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND10.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND10.xlsx'});
    elseif shiftedDays==11
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND11.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND11.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND11.xlsx'});
    elseif shiftedDays==12
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND12.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND12.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND12.xlsx'});
    end
    else
    filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys.xlsm'});
    filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated.xlsx'});
    filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {
```

```
' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated.xlsx'})
    ;
end
elseif strcmp(time_period,'daily')
filename = strcat('\Data\The BofA Merrill Lynch US Corporate Index', {' '
    }, year_selected, '.xlsm');
%cds_spreads = xlsread('\Data\US 1-3 ML yield Spread-daily.xlsx');
end

[data_defaults_xls,temps] = xlsread(filename1{1});
dates = datenum(temps(2:end,1));
num_defaults = data_defaults_xls(:,1);

%read number of names per month
num_names_per_month_xls = xlsread(filename2{1});
num_names_per_month = num_names_per_month_xls(:,1);

combined_matrix=zeros(size(num_defaults,1),2);
combined_matrix(:,1)=num_defaults;
combined_matrix(:,2)=num_names_per_month;

num=xlsread(filename{1});
%cds_spreads_2yr_1_3yr = cds_spreads(:,1)/100.0;
%cds_spreads_3yr_1_3yr = cds_spreads(:,2)/100.0;

yield_to_mty_vect_after= num(:,1)/100.0;     %@t+1
yield_to_mty_vect_before= num(:,3)/100.0;    %@t
duration_vect_after = num(:,2);              %@t
duration_vect_before = num(:,4);              %@t
govt_oas = num(:,5)/120000.0;                %@t+1
price_vect = num(:,6);

num_rows = size(yield_to_mty_vect_after, 1);

cds_spreads_time_vect = zeros(num_rows-1,1);

for i=1:num_rows-1 %option1
%for i=97:num_rows-1 %option2
cds_spreads_time_vect(i,1) = govt_oas(i,1);
end

% optimisation begins from here
options = optimset('Algorithm','interior-point','Display','iter','
    MaxFunEvals', 1e+10, 'TolFun', 1.0e-10);

if strcmp(modelchoice,'linear')
theta_1 = 1; % lambda_a
%theta_2 = 1; % lambda_b
thetas=theta_1;
lb=-Inf;
ub=Inf;
else
theta_1 = 1; % lambda_a
theta_2 = 1; % lambda_b
thetas=[theta_1,theta_2];
lb=[-Inf;-Inf];
ub=[Inf;Inf];
end

if strcmp(with_num_names, 'true')
[x_unc1,fval_unc1,exitflag_unc1,output_unc1,lambda_unc1,grad_unc1,
    hessian_unc1] = fmincon(@(x) poissonMLE_cPGM_S(x, combined_matrix,
    cds_spreads_time_vect,modelchoice),thetas,[],[],[],[],lb,ub,[],options
```

```
    );
else
[x_unc1,fval_unc1,exitflag_unc1,output_unc1,lambda_unc1,grad_unc1,
    hessian_unc1] = fmincon(@(x) poissonMLE_cPGM_S(x, num_defaults,
    cds_spreads_time_vect,modelchoice),thetas,[],[],[],[],lb,ub,[],options
    );
end

disp('x_unc1');
fprintf('\n%.12f\n',x_unc1);

disp('fval_unc1');
fprintf('\n%.12f\n',fval_unc1);

disp('grad_unc1');
fprintf('\n%.12f\n',grad_unc1);

disp('hessian_unc1');
fprintf('\n%.12f\n',hessian_unc1);
```

```
function [ minusLogLikelihoodF ] = poissonMLE_cPGM_S(theta,
    numDefaults_matrix, govtOASs, modelchoice)
%UNTITLED Summary of this function goes here
%   minus log likelihood function, explainatory variables excluding cds
%   spreads
% size of LAMBDA and K should be the same

if strcmp(modelchoice,'linear')
lambda_a = theta(1,1);
else
lambda_a = theta(1,1);
lambda_b = theta(1,2);
end

% check whether number of members per month is included
col_nums_defaults = size(numDefaults_matrix,2);
row_nums_defaults = size(numDefaults_matrix,1);
if( col_nums_defaults==2)
numDefaults = numDefaults_matrix(:,1);
num_names_per_month = numDefaults_matrix(:,2);
else
numDefaults = numDefaults_matrix;
num_names_per_month = ones(row_nums_defaults,1);
end

rows_num = size(govtOASs,1); % the first line of defaults is to be
    discarded
logLikelihoodF = 0;

for i=1:rows_num %option1
num_default = numDefaults(i,1); % using the row number of price data
num_names = num_names_per_month(i,1);
oas_sprd = govtOASs(i,1);
if (strcmp(modelchoice,'linearb'))
lambda = num_names*(lambda_a*oas_sprd+lambda_b);
elseif (strcmp(modelchoice,'linearb2'))
lambda = num_names*(lambda_a*oas_sprd^(2)+lambda_b);
elseif (strcmp(modelchoice,'linear'))
lambda = num_names*(lambda_a*oas_sprd);
elseif (strcmp(modelchoice,'linear2'))
lambda = num_names*(lambda_a*oas_sprd^(2));
elseif (strcmp(modelchoice,'loglinear1'))
lambda = num_names*(lambda_b*(oas_sprd^(lambda_a)));
```

```
        elseif (strcmp(modelchoice,'loglinear2'))
        lambda = num_names*exp(lambda_b+lambda_a*log(oas_sprd));
        end

        if (lambda<0)
        minusLogLikelihoodF=10000000000;
        return
        end

        logLikelihoodF = logLikelihoodF + num_default*log(lambda) - lambda - log(factorial(num_default));

        end

        minusLogLikelihoodF = -logLikelihoodF;

        end
```

A.2 Parameter estimation of the gamma distribution

We use the following Matlab function to estimate parameters of the gamma distribution $G(k_t\alpha, M_t\beta_s)$.

```
        year_selected = 'Master Year';
        time_period = 'monthly';
        with_cds_spread = 'false';

        %model configuration
        with_num_names = 'true';
        gammaBeta_modelchoice='loglineard12'; % nondependent, linear, lineard, loglinear12,loglinear32,loglineard12,loglineard32
        gamma_modelChoice='Model1'; %Model1=beta*number_of_names (used for the paper), Model2=gamma_alpha/number_of_names

        %Gamma model configuration
        residual_floor = 0.0;

        %shifting data configuration
        shifted_flag = 'true';
        shiftedDays = 2;

        if strcmp(time_period,'monthly')
        if strcmp(shifted_flag, 'true')
        if shiftedDays==1
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND1.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND1.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND1.xlsx'});
        elseif shiftedDays==2
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND2.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND2.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND2.xlsx'});
        elseif shiftedDays==3
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND3.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND3.xlsx'});
```

```
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND3.xlsx'});
        elseif shiftedDays==4
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND4.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND4.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND4.xlsx'});
        elseif shiftedDays==5
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND.xlsx'});
        elseif shiftedDays==6
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND6.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND6.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND6.xlsx'});
        elseif shiftedDays==7
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND7.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND7.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND7.xlsx'});
        elseif shiftedDays==8
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND8.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND8.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND8.xlsx'});
        elseif shiftedDays==9
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND9.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND9.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND9.xlsx'});
        elseif shiftedDays==10
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND10.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND10.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND10.xlsx'});
        elseif shiftedDays==11
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND11.xlsm'});
```

```
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND11.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND11.xlsx'});
        elseif shiftedDays==12
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys_MND12.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated_MND12.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated_MND12.xlsx'});
        end
        else
        filename = strcat({'\Data\The BofA Merrill Lynch US High Yield Index Master Year-monthly - MK_Data-SpreadsBetweenH1BUYTM-Treasury2Y_truncated2007-Ys.xlsm'});
        filename1 = strcat({'\Data\S&P_US_HY_NumDefaults_truncated.xlsx'});
        filename2 = strcat({'\Data\The BofA Merrill Lynch US High Yield Index'}, {' '}, year_selected, {'-monthly - MK_Data_numOfNames_truncated.xlsx'});
        end
        elseif strcmp(time_period,'daily')
        filename = strcat('\Data\The BofA Merrill Lynch US Corporate Index', {' '}, year_selected, '.xlsm');
        end

        [data_defaults_xls,temps] = xlsread(filename1{1});
        dates = datenum(temps(2:end,1));
        num_defaults = data_defaults_xls(:,1);

        %read number of names per month
        num_names_per_month_xls = xlsread(filename2{1});
        num_names_per_month = num_names_per_month_xls(:,1);

        combined_matrix=zeros(size(num_defaults,1),2); %2012b zeros(size(num_defaults),2);
        combined_matrix(:,1)=num_defaults;
        combined_matrix(:,2)=num_names_per_month;

        num=xlsread(filename{1},'Main');

        yield_to_mty_vect_after= num(:,1)/100.0;     %@t+1
        yield_to_mty_vect_before= num(:,3)/100.0;    %@t
        duration_vect_after = num(:,2);              %@t
        duration_vect_before = num(:,4);              %@t
        govt_oas = num(:,5)/120000.0;                %@t+1
        price_vect = num(:,6);

        num_rows = size(yield_to_mty_vect_after, 1);

        dur_ytm = zeros(num_rows-1,1);
        ytm_vect_diff = zeros(num_rows-1,1);
        ytm_vect_diff_square = zeros(num_rows-1,1);
        price_vect_log_diff = zeros(num_rows-1,1);
        residuals = zeros(num_rows-1,1);
        residuals_final = zeros(num_rows-1,1);

        floored_minus_res_final =  zeros(num_rows-1,1);
        cds_spreads_time_vect = zeros(num_rows-1,1);

        for i=1:num_rows-1 %option1
        %for i=97:num_rows-1 %option2
```

```
        ytm_vect_diff(i,1) = yield_to_mty_vect_before(i+1,1)-
            yield_to_mty_vect_after(i,1);
        price_vect_log_diff(i,1) = log(price_vect(i+1,1)) - log(price_vect(i,1));
        end

        ytm_vect_diff_square = 0.5*(ytm_vect_diff.^2);

        for i=1:num_rows-1 %option1
        %for i=97:num_rows-1 %option2
        if strcmp(time_period,'monthly') %used for offical paper
        if strcmp(with_cds_spread, 'true')
        % Method 1
        residuals(i,1) = price_vect_log_diff(i,1) - yield_to_mty_vect_before(i
            +1,1)/12.0 + cds_spread_vect(i+1,1)/12.0;
        else
        % Method1
        residuals(i,1) = price_vect_log_diff(i,1) - yield_to_mty_vect_before(i
            +1,1)/12.0;
        end
        elseif strcmp(time_period,'daily')
        if strcmp(with_cds_spread, 'true')
        residuals(i,1) = price_vect_log_diff(i,1) - yield_to_mty_vect_before(i
            +1,1)/365.0 + cds_spread_vect(i+1,1)/365.0;
        else
        residuals(i,1) = price_vect_log_diff(i,1) - yield_to_mty_vect_before(i
            +1,1)/365.0;
        end
        end

        cds_spreads_time_vect(i,1) = govt_oas(i,1);

        %Method 2
        dur_ytm(i,1) = duration_vect_before(i+1, 1)* ytm_vect_diff(i,1);
        residuals_final(i,1) = residuals(i,1) + dur_ytm(i,1);
        end

        minus_res_final = -residuals_final;
        disp('max -residuals');
        fprintf('%.12f\n',max(minus_res_final));

        figure;
        plot(minus_res_final,'b')
        hold('on')
        grid('on')
        plot(cds_spreads_time_vect, 'r')
        title(strcat('minus Residuals ', {', '}, year_selected, {' index, '},
            time_period));
        legend({'minus residuals', 'cds spread*time'}, 'Location', 'Best');

        %for i=1:num_rows-1 %option1
        for i=1:num_rows-1 %option2
        floored_minus_res_final(i,1) = max(minus_res_final(i,1), residual_floor);
            %0.000000000001
        end

        %--- plotting
        fig=figure;
        line1=plot(dates,floored_minus_res_final,'-b');
        hold('on')
        grid('on')
        line2=plot(dates,cds_spreads_time_vect, '--r');

        hold('off')
```

```
        datetick('x','yyyy/mm','keeplimits', 'keepticks')
        h_legend=legend([line1,line2],{char('Unexplained Negative Return Residuals of the Single Factor Model'),char('Spread between H1BU YTM and YTM of On-the-Run 2Y US Treasury Note')},'Location','southoutside');
        set(h_legend,'FontSize',8);

        xlabel('Dates');
        ylabel('Unexplained Negatgive Return Residuals/Spreads');
        title('Spread Study');
        xlim([min(dates) max(dates)])
        disp(fig)
        filename1='C:\Hong document\Academics\PhD\Homework\2017-08-23\Spreads\SpreadsStudy.png';
        saveas(gcf,filename1);

        fig1=figure;
        set(fig1, 'Visible', 'on');
        [hAx,hLine1,hLine2]=plotyy(dates,floored_minus_res_final, dates,
            num_defaults);
        grid('on');
        set(hAx, 'xTickLabel','')
        datetick('x','yyyy/mm','keeplimits', 'keepticks')
        h_legend=legend([hLine1,hLine2],{char('Unexplained Negative Return Residuals of the Single Factor Model'),char('S&P US HY Number of Defaults')},'Location','southoutside');
        set(h_legend,'FontSize',8);

        xlabel('Dates');
        title('Number of Defaults Study');
        ylabel(hAx(1),'Unexplained Negative Return Residuals');
        ylabel(hAx(2),'S&P US HY Number of Defaults');
        set(hAx(1),'YColor','b');
        set(hAx(2),'YColor','r');
        set(hLine1,'color','b');
        set(hLine2,'color','r');
        set(hLine1,'LineStyle','-');
        set(hLine2,'LineStyle','--');
        xlim(hAx(1),[min(dates) max(dates)]);
        xlim(hAx(2),[min(dates) max(dates)]);

        disp(fig1);
        filename2='C:\Hong document\Academics\PhD\Homework\2018-01-21\Spreads\ShiftedDefaultStudy.png';
        saveas(gcf,filename2);
        %----



        % optimisation begins from here
        options = optimset('Algorithm','interior-point','MaxFunEvals', 1e+100, 'TolFun', 1.0e-6);

        if (strcmp(gammaBeta_modelchoice,'lineard') || ~isempty(strfind(
            gammaBeta_modelchoice,'loglinear')))
        theta_3 = 0.1; % gamma_alpha has to be bigger than 0
            floored_minus_res_final
        theta_4 = 0.1; % gamma_beta_c  gamma_beta_c * S_{t} + gamma_beta_d>0
        theta_5 = 0.1; % gamma_beta_d

        thetas=[theta_3,theta_4,theta_5];
        lb=[0.00000000001;-Inf;-Inf];
        ub=[Inf;Inf;Inf];
        else
        theta_3 = 0.1; % gamma_alpha has to be bigger than 0
```

```
    floored_minus_res_final
theta_4 = 0.1; % gamma_beta has to be bigger than 0

thetas=[theta_3,theta_4];
lb=[0.00000000001;0.00000000001];
ub=[Inf;Inf];
end

if strcmp(with_num_names, 'true')
[x_unc1,fval_unc1,exitflag_unc1,output_unc1,lambda_unc1,grad_unc1,
    hessian_unc1] = fmincon(@(x) gammaMLE_cPGM_S(x, combined_matrix,...
floored_minus_res_final, cds_spreads_time_vect, gamma_modelChoice,
    gammaBeta_modelchoice),thetas,[],[],[],[],lb,ub,[],options);
else
[x_unc1,fval_unc1,exitflag_unc1,output_unc1,lambda_unc1,grad_unc1,
    hessian_unc1] = fmincon(@(x) gammaMLE_cPGM_S(x, num_defaults, ...
floored_minus_res_final, cds_spreads_time_vect, gamma_modelChoice,
    gammaBeta_modelchoice),thetas,[],[],[],[],lb,ub,[],options);
end

disp('x_unc1');
fprintf('%14.12e\n',x_unc1);

disp('fval_unc1');
fprintf('\n%14.12e\n',fval_unc1);

disp('grad_unc1');
fprintf('\n%14.12e\n',grad_unc1);

disp('hessian_unc1');
fprintf('\n%14.12e\n',hessian_unc1);
```

```
function [ minusLogLikelihoodF ] = gammaMLE_cPGM_S( theta, defaults_matrix
    , residuals_adj, govtOASs, gamma_modelChoice, gammaBeta_modelchoice )
%UNTITLED Summary of this function goes here
% minus log likelihood function
% theta_1 = K(i,1)*alpha-1
% theta_2 = -beta
% alpha is the shape parameter of the gamma distribution
% beta is the scale parameter of the gamma distrubtion
%   Detailed explanation goes here

if (strcmp(gammaBeta_modelchoice,'lineard') || ~isempty(strfind(
    gammaBeta_modelchoice,'loglinear')))
gamma_alpha = theta(1,1);
gamma_beta_c = theta(1,2);
gamma_beta_d = theta(1,3);
else
gamma_alpha = theta(1,1);
gamma_beta = theta(1,2);
end

% check whether number of members per month is included
col_nums_defaults = size(defaults_matrix,2);
row_nums_defaults = size(defaults_matrix,1);
if( col_nums_defaults== 2)
defaults = defaults_matrix(:,1);
num_names_per_month = defaults_matrix(:,2);
else
defaults = defaults_matrix;
num_names_per_month = ones(row_nums_defaults,1);
end

```

```
    rows_num = size(residuals_adj,1);
    logLikelihoodF = 0;

    for i=1:rows_num %option1
    num_default = defaults(i,1);
    num_names = num_names_per_month(i,1);
    oas_sprd = govtOASs(i,1);

    if (strcmp(gammaBeta_modelchoice,'linear'))
    beta_adjusted = (gamma_beta/oas_sprd);
    elseif (strcmp(gammaBeta_modelchoice,'nondependent'))
    beta_adjusted = gamma_beta;
    elseif (strcmp(gammaBeta_modelchoice,'lineard'))
    beta_adjusted = (gamma_beta_c/oas_sprd + gamma_beta_d);
    elseif (strcmp(gammaBeta_modelchoice,'loglinear12'))
    beta_adjusted = gamma_beta_c/power(oas_sprd, 1/2);
    elseif (strcmp(gammaBeta_modelchoice,'loglinear32'))
    beta_adjusted = gamma_beta_c/power(oas_sprd, 3/2);
    elseif (strcmp(gammaBeta_modelchoice,'logylineard12'))
    beta_adjusted = gamma_beta_c/power(oas_sprd, 1/2)+gamma_beta_d;
    elseif (strcmp(gammaBeta_modelchoice,'loglineard32'))
    beta_adjusted = gamma_beta_c/power(oas_sprd, 3/2)+gamma_beta_d;
    end

    if(strcmp(gamma_modelChoice,'Model1'))
    beta_adjusted_adj=beta_adjusted*num_names;
    gamma_alpha_adj=gamma_alph
    elseif(strcmp(gamma_modelChoice,'Model2'))
    beta_adjusted_adj=beta_adjusted;
    gamma_alpha_adj=gamma_alpha/num_names;
    end

    if (beta_adjusted_adj<=0) || (gamma_alpha_adj<=0)
    minusLogLikelihoodF=10000000000;
    return
    end

    if(residuals_adj(i,1)>0)
    k_alpha = num_default*gamma_alpha_adj;
    if(k_alpha==0)
    continue;
    end
    residuals_adj_M=residuals_adj(i,1);

    increment=k_alpha*log(beta_adjusted_adj) + (k_alpha-1)*log(residuals_adj_M)-beta_adjusted_adj*residuals_adj_M-log(gamma(k_alpha));

    logLikelihoodF = logLikelihoodF + increment;
    disp('i');
    fprintf('\n%.12f\n',i);
    disp('logLikelihoodF');
    fprintf('\n%14.6e\n',logLikelihoodF);
    end
    end

    minusLogLikelihoodF = -logLikelihoodF;

    end
```

Appendix B Algorithms and Matlab code for simulation

B.1 Inhomogeneous Poisson distribution simulation

B.1.1 Simulation algorithm

We use the following algorithm simulate the inhomogeneous Poisson distribution $Pois(M_t \lambda_s(s-t))$ over time for one simulation.

Step 1 Initialize $T := 1; t := 0; N = 0;$

Step 2 Repeat following steps until $t > T$

Step 3 Generate a random number u uniformly distributed over $[0, 1]$.

Step 4 Set $t := t - \dfrac{\ln u}{\lambda_s(s-t)}$

Step 5 Set $N := N + 1$

B.1.2 Matlab code

The above simulation algorithm is written in the Matlab function below to simulate the inhomogeneous Poisson distribution
$Pois(M_t \lambda_s(s-t))$ for the whole observation period.

```
function [ pdfs, means, medians, percQs_10, percQs_90, confI_90L,confI_90H ] = PoissonPdf2( num_rows, number_sims, cdsSpreads ...
,numNamePerMonth, lambda_a, lambda_b, lambda_modelchoice, includedNumNamesPerMonth)
% This function returns the pdf of compound Poisson Gamma distribution
%   Detailed explanation goes here

Poiss_pdfs = zeros(num_rows, 1);
Poiss_means = zeros(num_rows, 1);
Poiss_medians = zeros(num_rows, 1);
Poiss_quantiles_90perc = zeros(num_rows, 1);
Poiss_quantiles_10perc = zeros(num_rows, 1);
cond_Interval_90_lower = zeros(num_rows, 1);
cond_Interval_90_higher = zeros(num_rows,1);


% loop throught each month
% a distribution per month
for i=1:num_rows %default data option 1
govt_spread = cdsSpreads(i,1);

if (strcmp(lambda_modelchoice, 'linearb'))
lambda_ind = lambda_a * govt_spread  + lambda_b;
elseif (strcmp(lambda_modelchoice, 'linearb2'))
lambda_ind = lambda_a * govt_spread^(2)+lambda_b;
elseif (strcmp(lambda_modelchoice, 'linear'))
lambda_ind = lambda_a * govt_spread;
elseif (strcmp(lambda_modelchoice, 'linear2'))
lambda_ind = lambda_a * govt_spread^(2);
elseif (strcmp(lambda_modelchoice, 'loglinear1'))
lambda_ind = lambda_b * govt_spread^(lambda_a);
elseif (strcmp(lambda_modelchoice, 'loglinear2') || strcmp(
    lambda_modelchoice, 'average'))
lambda_ind = exp(lambda_b + lambda_a*log(govt_spread));
end

if (strcmp(includedNumNamesPerMonth, 'true'))
num_names = numNamePerMonth(i,1);
lambda = num_names * lambda_ind;
```

```
        elseif (strcmp(includedNumNamesPerMonth, 'false'))
        lambda = lambda_ind;
        end

        if( lambda <0 )
        disp('lambda smaller than 0');
        lambda=0;
        end

        % simulating Poisson.

        num_sims = number_sims;
        T = 1; % 1 month

        sampled_N = zeros(num_sims,1);
        s1= RandStream.create('mrg32k3a','NumStreams',num_rows, 'StreamIndices', i
            ,'Seed', 'shuffle');
        RandStream.setGlobalStream(s1);

        for j_sim=1:num_sims
        count_t = 0;
        count_events = 0;
        while count_t < T
        rand_num = rand(s1, 1, 1);
        count_t = count_t + (-(1/lambda)*log(rand_num));
        if( count_t > T )
        break;
        else
        count_events = count_events + 1;
        end
        end
        sampled_N(j_sim,1) = count_events;
        end

        sampled_size=sampled_N;
        sorted_sampled_size = sort(sampled_size,'descend');
        Poiss_means(i,1) = mean(sorted_sampled_size,1);

        if(num_sims>1)
        num_90th = floor(0.9*num_sims); % pick the 10th biggest number
        num_10th = floor(0.1*num_sims); % pick the 90th biggest number
        middle_num_floor = floor(0.5*num_sims);
        middle_num_ceil = ceil(0.5*num_sims);
        median = 0.5*(sorted_sampled_size(middle_num_floor,1) +
            sorted_sampled_size(middle_num_ceil,1));
        Poiss_medians(i,1) = median;
        Poiss_quantiles_90perc(i,1) = sorted_sampled_size(num_90th, 1);
        Poiss_quantiles_10perc(i,1) = sorted_sampled_size(num_10th, 1);
        sigma=sqrt(var(sorted_sampled_size,1));
        cond_Interval_90_lower(i,1)=Poiss_means(i,1)-sigma*1.645;
        cond_Interval_90_higher(i,1)=Poiss_means(i,1)+sigma*1.645;
        end
        end

        pdfs = Poiss_pdfs;
        means = Poiss_means;
        medians = Poiss_medians;
        percQs_10 = Poiss_quantiles_10perc;
        percQs_90 = Poiss_quantiles_90perc;
        confI_90L = cond_Interval_90_lower;
        confI_90H = cond_Interval_90_higher;

        end
```

B.2 Gamma distribution simulation

We use Matlab function *gamrnd* to generate the gamma random numbers. The following Matlab function is used to simulate the gamma distribution $G(k_t\alpha, M_t\beta_s)$ for the whole observation period.

```
function [ pdfs, means, medians, percQs_10, percQs_90, confI_90L, confI_90H] = gammaPdf( num_rows, number_sims, cdsSpreads, ...
numNamePerMonth, gamma_alpha, gamma_betas, gammaBeta_modelchoice, gammaModelChoice, includedNumNamesPerMonth, numDefaults)
% This function returns the pdf of compound Poisson Gamma distribution
%  Detailed explanation goes here

gamma_pdfs = zeros(num_rows, 1);
gamma_means = zeros(num_rows, 1);
gamma_medians = zeros(num_rows, 1);
gamma_quantiles_90perc = zeros(num_rows, 1);
gamma_quantiles_10perc = zeros(num_rows, 1);
cond_Interval_90_lower = zeros(num_rows, 1);
cond_Interval_90_higher = zeros(num_rows,1);

num_sims = number_sims;

% loop throught each month
% a distribution per month
for i_months=1:num_rows
govt_spread = cdsSpreads(i_months,1);
numDefault=numDefaults(i_months,1);
gamma_alpha_adj=gamma_alpha*numDefault;

if (strcmp(gammaBeta_modelchoice, 'nondependent'))
gamma_beta = gamma_betas(1,1);
gamma_beta_adj = gamma_beta;
elseif (strcmp(gammaBeta_modelchoice, 'linear'))
gamma_beta = gamma_betas(1,1);
gamma_beta_adj = gamma_beta/govt_spread;
elseif (strcmp(gammaBeta_modelchoice, 'lineard'))
gamma_beta_c = gamma_betas(1,1);
gamma_beta_d = gamma_betas(1,2);
gamma_beta_adj = gamma_beta_c/govt_spread + gamma_beta_d;
elseif (strcmp(gammaBeta_modelchoice, 'loglinear12') || strcmp(gammaBeta_modelchoice, 'loglineard12') || strcmp(gammaBeta_modelchoice, 'average'))
gamma_beta_c = gamma_betas(1,1);
gamma_beta_d = gamma_betas(1,2);
gamma_beta_adj = gamma_beta_c/power(govt_spread, 1/2) + gamma_beta_d;
elseif (strcmp(gammaBeta_modelchoice, 'loglinear32') || strcmp(gammaBeta_modelchoice, 'loglineard32'))
gamma_beta_c = gamma_betas(1,1);
gamma_beta_d = gamma_betas(1,2);
gamma_beta_adj = gamma_beta_c/power(govt_spread, 3/2) + gamma_beta_d;
end

if (strcmp(includedNumNamesPerMonth, 'true'))
num_names = numNamePerMonth(i_months,1);
if(strcmp(gammaModelChoice,'Model1'))
beta_adjusted = num_names * gamma_beta_adj;
alpha_adjusted=gamma_alpha_adj;
elseif(strcmp(gammaModelChoice,'Model2'))
beta_adjusted=gamma_beta_adj;
alpha_adjusted = gamma_alpha_adj/num_names;
end
elseif (strcmp(includedNumNamesPerMonth, 'false'))
beta_adjusted = gamma_beta_adj;
alpha_adjusted = gamma_alpha_adj;
```

```
        end

        %       alpha_adjusted = num_names * gamma_alpha; %gamma_alpha * k; %not for simulating

        s1= RandStream.create('mrg32k3a','NumStreams',num_rows, 'StreamIndices', i_months, 'Seed', 'shuffle'); %'shuffle'
        RandStream.setGlobalStream(s1);

        sampled_size = zeros(num_sims, 1);

        for j_sim=1:num_sims
        sampled_size(j_sim,1) = gamrnd(alpha_adjusted, 1/beta_adjusted);
        if(sampled_size(j_sim,1)<0)
        disp('wrong gamma random varialbes!!!')
        end
        end

        sorted_sampled_size = sort(sampled_size,'descend');
        gamma_means(i_months,1) = mean(sorted_sampled_size,1);

        if(num_sims>1)
        num_90th = floor(0.9*num_sims); % pick the 5th biggest number
        num_10th = floor(0.1*num_sims); % pick the 95th biggest number
        middle_num_floor = floor(0.5*num_sims);
        middle_num_ceil = ceil(0.5*num_sims);
        median = 0.5*(sorted_sampled_size(middle_num_floor,1) + sorted_sampled_size(middle_num_ceil,1));
        gamma_medians(i_months,1) = median;
        gamma_quantiles_90perc(i_months,1) = sorted_sampled_size(num_90th, 1);
        gamma_quantiles_10perc(i_months,1) = sorted_sampled_size(num_10th, 1);
        sigma=sqrt(var(sorted_sampled_size,1));
        cond_Interval_90_lower(i_months,1)=gamma_means(i_months,1)-sigma*1.645;
        cond_Interval_90_higher(i_months,1)=gamma_means(i_months,1)+sigma*1.645;
        end
        end

        pdfs = gamma_pdfs;
        means = gamma_means;
        medians = gamma_medians;
        percQs_10 = gamma_quantiles_10perc;
        percQs_90 = gamma_quantiles_90perc;
        confI_90L = cond_Interval_90_lower;
        confI_90H = cond_Interval_90_higher;
        end
```

B.3 Compound (inhomogeneous) Poisson gamma distribution simulation

B.3.1 Simulation algorithm

To simulate the compound (inhomogeneous) Poisson gamma distribution, both the inhomogeneous Poisson and the gamma distributions must be simulated. We use the following algorithm to simulate the compound (inhomogeneous) Poisson gamma distribution $CPG(M_t\lambda_s(s-t), \alpha, M_t\beta_s)$ for one simulation (Cont and Tankov (2004).).

Step 1 Initialize $T := 1; t := 0; N := 0; L := 0;$

Step 2 Repeat following steps until $t > T$

Step 3 Generate a random number u uniformly distributed over $[0, 1]$.

Step 4 Set $t := t - \dfrac{\ln u}{\lambda_s(s-t)}$

Step 5 Set $L := L + gamrnd(\alpha_t, \frac{1}{\beta_s})$

Step 6 Set $N := N + 1$

The function *gamrnd* is the Matlab function used to generate gamma random variables.

B.3.2 Matlab code

The above simulation algorithm is written in the Matlab function below to simulate the compound (inhomogeneous) Poisson gamma distribution $CPG(M_t\lambda_s(s-t), \alpha_t, M_t\beta_s)$ for the whole observation period.

```
function [ pdfs, means, medians, percQs_10, percQs_90,confI_90L, confI_90H
    ] = compoundPoissonPdf( num_rows, number_sims, cdsSpreads, ...
numNamePerMonth, lambda_a, lambda_b, gamma_alpha, gamma_betas,
    lambda_modelchoice, gammaBeta_modelchoice, gammaModelChoice,...
includedNumNamesPerMonth, includedNumNamesPerMonth_Poisson)
% This function returns the pdf of compound Poisson Gamma distribution
%   Detailed explanation goes here

%num_rows = size(num_rows,1);

compoundPoiss_pdfs = zeros(num_rows, 1);
compoundPoiss_means = zeros(num_rows, 1);
compoundPoiss_medians = zeros(num_rows, 1);
compoundPoiss_quantiles_95perc = zeros(num_rows, 1);
compoundPoiss_quantiles_10perc = zeros(num_rows, 1);
cond_Interval_90_lower = zeros(num_rows, 1);
cond_Interval_90_higher = zeros(num_rows,1);

% loop throught each month
% a distribution per month
for i=1:num_rows %default data option 1
if (strcmp(lambda_modelchoice, 'linearb'))
lambda_ind = lambda_a * govt_spread  + lambda_b;
elseif (strcmp(lambda_modelchoice, 'linearb2'))
lambda_ind = lambda_a * govt_spread^(2)  + lambda_b;
elseif (strcmp(lambda_modelchoice, 'linear'))
lambda_ind = lambda_a * govt_spread;
elseif (strcmp(lambda_modelchoice, 'linear2'))
lambda_ind = lambda_a * govt_spread^(2);
elseif (strcmp(lambda_modelchoice, 'loglinear2') || strcmp(
    lambda_modelchoice, 'average'))
lambda_ind = exp(lambda_b+lambda_a*log(govt_spread));
elseif (strcmp(lambda_modelchoice, 'loglinear1'))
lambda_ind = lambda_b*(govt_spread^(lambda_a));
end

if (strcmp(gammaBeta_modelchoice, 'nondependent'))
gamma_beta = gamma_betas(1,1);
gamma_beta_adj = gamma_beta;
elseif (strcmp(gammaBeta_modelchoice, 'linear'))
gamma_beta = gamma_betas(1,1);
gamma_beta_adj = gamma_beta/govt_spread;
elseif (strcmp(gammaBeta_modelchoice, 'lineard'))
gamma_beta_c = gamma_betas(1,1);
gamma_beta_d = gamma_betas(1,2);
gamma_beta_adj = gamma_beta_c/govt_spread + gamma_beta_d;
elseif (strcmp(gammaBeta_modelchoice, 'loglinear32') || strcmp(
    gammaBeta_modelchoice, 'loglineard32'))
gamma_beta_c = gamma_betas(1,1);
gamma_beta_d = gamma_betas(1,2);
gamma_beta_adj = gamma_beta_c/power(govt_spread,3/2) + gamma_beta_d;
```

```
        elseif (strcmp(gammaBeta_modelchoice, 'loglinear12') ||        strcmp(
            gammaBeta_modelchoice, 'logleard12') || strcmp(gammaBeta_modelchoice
            , 'average'))
        gamma_beta_c = gamma_betas(1,1);
        gamma_beta_d = gamma_betas(1,2);
        gamma_beta_adj = gamma_beta_c/power(govt_spread,1/2) + gamma_beta_d;
        end

        if (strcmp(includedNumNamesPerMonth, 'true'))
        num_names = numNamePerMonth(i,1);
        if(strcmp(includedNumNamesPerMonth_Poisson,'true'))
        lambda = num_names * lambda_ind;
        else
        lambda = lambda_ind;
        end
        if(strcmp(gammaModelChoice,'Model1'))
        beta_adjusted = num_names * gamma_beta_adj;
        alpha_adjusted=gamma_alpha;
        elseif(strcmp(gammaModelChoice,'Model2'))
        beta_adjusted=gamma_beta_adj;
        alpha_adjusted = gamma_alpha/num_names;
        end
        elseif (strcmp(includedNumNamesPerMonth, 'false'))
        lambda = lambda_ind;
        beta_adjusted = gamma_beta_adj;
        alpha_adjusted= gamma_alpha;
        end

        if( lambda <0 )
        disp('lambda smaller than 0');
        break
        end

        % simulating Compound Poisson.

        num_sims = number_sims;
        T = 1; % 1 month

        sampled_N = zeros(num_sims,1);
        sampled_size = zeros(num_sims, 1);

        s1= RandStream.create('mrg32k3a','NumStreams',num_rows, 'StreamIndices', i
            ,'Seed', 'shuffle');
        RandStream.setGlobalStream(s1);

        for j_sim=1:num_sims
        count_t = 0;
        count_events = 0;
        sum_size = 0;
        while count_t < T
        rand_num = rand(s1, 1, 1);
        count_t = count_t + (-(1/lambda)*log(rand_num));
        if( count_t > T )
        break;
        else
        if (strcmp(includedNumNamesPerMonth, 'false'))
        gamma_rand_num = gamrnd(alpha_adjusted, 1/beta_adjusted);
        elseif(strcmp(includedNumNamesPerMonth, 'true'))
        gamma_rand_num = gamrnd(alpha_adjusted, 1/beta_adjusted);
        end
        count_events = count_events + 1;
        sum_size = sum_size + gamma_rand_num;
        end
```

```
        end
        sampled_N(j_sim,1) = count_events;
        sampled_size(j_sim,1) = sum_size;
        end

        sorted_sampled_size = sort(sampled_size,'descend');
        compoundPoiss_means(i,1) = mean(sorted_sampled_size,1);

        if(num_sims>1)
        num_95th = floor(0.95*num_sims); % pick the 10th biggest number
        num_05th = floor(0.05*num_sims); % pick the 90th biggest number
        %middle_num_floor = floor(0.5*num_sims);
        %middle_num_ceil = ceil(0.5*num_sims);
        num_50th=floor(0.5*num_sims);
        %median = 0.5*(sorted_sampled_size(middle_num_floor,1) +
            sorted_sampled_size(middle_num_ceil,1));
        compoundPoiss_medians(i,1) = sorted_sampled_size(num_50th, 1);
        compoundPoiss_quantiles_95perc(i,1) = sorted_sampled_size(num_95th, 1);
        compoundPoiss_quantiles_10perc(i,1) = sorted_sampled_size(num_05th, 1);
        end
        end

        pdfs = compoundPoiss_pdfs;
        means = compoundPoiss_means;
        medians = compoundPoiss_medians;
        percQs_10 = compoundPoiss_quantiles_10perc;
        percQs_90 = compoundPoiss_quantiles_95perc;
        confI_90L = cond_Interval_90_lower;
        confI_90H = cond_Interval_90_higher;
        end
```

References

[1] Kenneth P. Burnham and David R. Anderson. *Model Selection and MultiModel inference: A Practical Information-Theoretic Approach, Second Edition.* Springer, 2002.

[2] Pierre Collin-Dufresne and Bruno Solnik. *On the Term Structure of Default Premia in the Swap and LIBOR Markets.* The Journal of Finance, 56, 1095-2015, 2001.

[3] Rama Cont and Peter Tankov. *Financial Modelling With Jump Processes.* CRC Press LLC, 2004.

[4] Joost Driessen. *Is Default Event Risk Priced in Corporate Bonds?* The Review of Financial Studies, Vol. 18, No. 1 (Spring, 2005), pp. 165-195., 2005.

[5] Gregory R. Duffee. *Estimating the Price of Default Risk.* The Review of Financial Studies, 12, 197-226., 2005.

[6] Darrell Duffie and Kenneth J. Singleton. *An Econometric Model of the Term Structure of Interest Rate Swap Yields.* The Journal of Finance Vol. 52, No. 4 (Sep., 1997), pp. 1287-1321, 1997.

[7] Darrell Duffie and Kenneth J. Singleton. *Modelling term structures of defaultable bonds.* The Review of Financial Studies, 12(4):687-720,1999, 1999.

[8] John Hull and Alan White. *The Impact of Default Risk on Option Prices and empirical implications.* Journal of Banking & Finance, Vol. 19, No. 2, (May 1995), pp. 299-322, 1995.

[9] A. Ilmanen, D. McGuire, and A. Warga. *The Value of Duration as A Risk Measure for Corporate Debt.* The Journal of Fixed Income, 4(1):70-76, 1994, 1994.

[10] Robert A. Jarrow. *The Relationship Between Yield, Risk, and Return of Corporate Bonds.* The Journal of Finance, Vol. 33, No. 4 (Sep., 1978), pp. 1235-1240, 1978.

[11] Robert A. Jarrow, David Lando, and Stuart M. Turnbull. *A Markov Model for the Term Structure of Credit Risk Spread.* The Review of Financial Studies Summer 1997 Vol. 10, No. 2, pp. 481–523, 1997.

[12] Robert A. Jarrow, David Lando, and Fan Yu. *Default risk and diversification: Theory and Empirical Implications.* Mathematical Finance, 15(1):1-26, 2005, 2005.

[13] Matti Koivu and Teemu Pennanen. *Reduced form models of bond portfolios.* Working paper, 2010.

[14] Matti Koivu and Teemu Pennanen. “Return Dynamics of Index-Linked Bond Portfolios”. In: *The Journal of Portfolio Management* 41.1 (Fall 2014), pp. 78–84.

[15] David Lando. *On Cox Processes and Credit-Risky Securities.* Review of Derivatives Research, 2:99-120, 1998., 1998.

[16] Jun Liu, Francis A. Longstaff, and Ravit E. Mandell. *The Market Price of Credit Risk: An Empirical Analysis of Interest Rate Swap Spreads.* 2000.

[17] Francis A. Longstaff and Eduardo S. Schwartz. “A Simple Approach to Valuing Risky Fixed and Floating Rate Debt”. In: *The Journal of Finance* 50.3 (July 1995), pp. 789–819.

[18] Dilip B. Madan and Haluk Unal. *Pricing the risks of default.* Review of Derivatives Research,December 1998, Vol 2, Issue 2–3, pp 121–160., 1998.

[19] Robert C. Merton. *On the Pricing of Corporate Debt: The Risk Structure of Interest Rates.* The Journal of Finance, Vol. 29, pp 449-470, 1974.

[20] Philipp J. Schönbucher. *Term structure modelling of defaultable bonds.* Review of Derivatives Research, Vol.2, Issue 2–3, pp. 161–192, 1998.

[21] Philipp J. Schönbucher. *Modeling Expected Return on Defaultable Bonds.* The Journal of Fixed Income, 12 (2) 69-81, 2002.

[22] Philipp J. Schönbucher. *Credit Derivatives Pricing Models: Model, Pricing and Implementation.* John Wiley & Sons Ltd, West Sussex, 2003.